DISCOVERING SCIENCE ON YOUR ADAM™ WITH 25 PROGRAMS

TALCOTT MOUNTAIN SCIENCE CENTER

DISCOVERING SCIENCE ON YOUR ADAM™ WITH 25 PROGRAMS

DISCOVERING SCIENCE ON YOUR ADAM™ WITH 25 PROGRAMS

BY THE TALCOTT MOUNTAIN SCIENCE CENTER

John Pellino
Mary Ellen Adamo
Sandra Dobrowlsky
Donald P. LaSalle, Ph.D.

TAB BOOKS Inc.
BLUE RIDGE SUMMIT, PA 17214

FIRST EDITION

FIRST PRINTING

Library of Congress Cataloging in Publication Data

Main entry under title:

Discovering science on your ADAM.

Includes index.
1. Science—Experiments—Data processing. 2. Adam (Computer)—Programming. 3. Basic (Computer program language) 4. Computer programs. I. Talcott Mountain Science Center. II. Title: Discovering science on your A.D.A.M.

Q182.3.D57 1984 502.8'5425 84-8719
ISBN 0-8306-0780-3
ISBN 0-8306-1780-9 (pbk.)

Cover illustration by Larry Selman.

Contents

	Acknowledgments	vi
	For Kids Only	vii
	For Parents Only	viii
1	**Adventure Preparations** Getting Started—Editing Programs—Saving Programs—Loading Programs—Entering Programs—Rules of Syntax—Numbers on the Screen—Screen Formatting—Remarks and Explanations—Catalog Program	1
2	**Adventures in Astronomy and Space Science** Planetary Distances and Diameters—Satellite Orbital Periods—Weight on Different Planets—Escape Velocity—Distance to the Sun	7
3	**Adventures in Earth Science** A Geological Time Clock—Rock Identification—Cloud Identification—Twelve-Hour Forecast—Measuring the Earth's Circumference	41
4	**Adventures in Life Science** Supersonic Bees—Evergreen Key—Population Simulation—Biology Matching Quiz	81
5	**Adventures in Mathematics and Computer Science** Base Converter—Height Finder—Scientific Notation—Alphabetic Order—Pretend Data	107
6	**Adventures in Physics** Frequency and Wavelength—Speed of Light—Pendulums—Metric Conversions	139
	Glossary	171
	Index	178

Acknowledgments

We wish to express our gratitude to those who helped in the preparation of these adventures:

To our fellow staff members Joe Caruso, Martin Sollomon, Dr. Nokos Solounias, Mark D'Antonio, Dr. Joe Bronzino of Trinity College: their technical advice was invaluable in helping to create this book.

To the teaching and research staff at Talcott Mountain Science Center who contributed programs, advice, and support.

To our administrators for their help in encouraging, organizing, and editing our work.

And finally to our families for their support in the midst of long nights and weekends and for keeping lots of dinners warm.

Thank you one and all.

For Kids Only

You're about to embark on a new adventure. Like all adventures, the fun lies in discovery—the finding new things, seeing new places, meeting new people. By using this book, you'll measure the size of the earth, tour the solar system, forecast weather, identify rocks, and analyze experiments. You'll visit all nine planets and rub shoulders with Newton, Einstein, Galileo, Darwin, Kepler, and supersonic bees!

All of this fun has the proper name of science—the process of doing your best with your brain to find out about the world around you. We hope that by the time you get through this book, you will see why scientists get so excited about their work. The intrigue of discovery and investigation, the satisfaction of working out solutions, and the breathtaking view you have while standing on the shoulders of giants is quite thrilling.

We said that this is an adventure, a journey of sorts. Your ADAM is your vehicle, this book is a map, and your imagination is your fuel, as well as your only limit. Go through your adventures as fast as you want. You can always backtrack and spend as much time as you like on any one of them. May all your adventures be happy ones!

For Parents Only

This book was written to give students a taste of science while becoming experienced in the use of a computer. We hope that it helps make these experiences both personally enjoyable and educationally productive.

We have drawn on our combined expertise in the fields of science and computers and, when necessary, called on master teachers and scientists to aid in the preparation of the text. This book represents over three decades of combined experience in education, science, and computers.

Talcott Mountain Science Center is dedicated to bringing science and its wonders to all students, by whatever means needed. In recent years, this has meant not only training students and teachers in the classroom, but also in sharing our expertise by means of television, computer software, and books such as this one.

As educators, our guidelines include some key concepts: individualized instruction and hands-on involvement. We feel that computer books can be excellent in this regard. Students learn one-on-one with the computer as a guide and may, as time goes by, involve friends, both around the block and around the world, to share in their adventures. Most of the programs contain experiments that have the student *doing* science, instead of simply reading about it or watching it done by someone else.

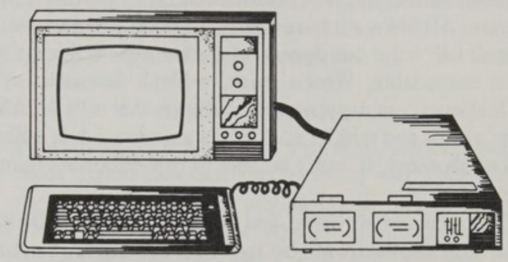
An ancient Chinese proverb sums this up quite nicely:

I hear, and I forget.
I see, and I remember.
I do, and I understand.

You're invited on these adventures, too. If you ask politely, we're sure your youngster will let you "tag along." (If not, tell them that you have our permission.)

Our best wishes for your child's successful adventures in science.

1



Adventure Preparations

All of the programs in this book were written for the ADAM computer using SmartBASIC This is the language that came with your machine. Since it is absolutely necessary to have SmartBASIC working correctly for these programs, we suggest that you save backup copy of SmartBASIC on a blank datapack. See your ADAM Owner's Manual for details.

Once SmartBASIC is up and running, you need only type in the program at the end of each adventure section, read the description of the adventure for some background, and run the program.

Most of the programs have some experiments for you and your friends to do. Others will simply ask you some questions. Many of the programs can be modified to run the way you'd like them to, or to do different things, or to improve upon them.

To change the programs, you need to be able to program in SmartBASIC. You may have learned some BASIC language at school or on a friend's computer. Chances are, if you can write in any BASIC, you can write in SmartBASIC, since they have a lot in common. If your BASIC skills need some polishing, check out the SmartBASIC manual that came with your ADAM, or stop by your local library or bookstore.

GETTING STARTED

Your ADAM comes with a printer, memory console, keyboard, game controllers, tv interface, manuals, and tapecassettes. You will need two of these cassettes: the SmartBASIC cassette and the blank digital datapack.

The Datapacks are Delicate. They use the same medium (magnetic coating) as audio tapes, but you must take better care of them. All magnetic media gets damaged in some way due to handling. This is no big deal for your musical tapes, because your ear is very tolerant, and a speck of static is small compared to the size of the musical notes. But that same speck on an ADAM tape can be disastrous. ADAM's ears are much more critical than yours, and one speck can wipe out dozens of data notes, making the whole program unreadable. What's more, a blank datapack isn't really blank. It always contains a road map of sorts that tells ADAM where the tape's ends and middle are. If this gets erased, it's like giving ADAM a lobotomy. It won't be able to remember anything on the tape, ever.

When you are through, and want to put the ADAM away, **remove any tapes from the tape drives before turning the printer off.** This has proven to be a leading cause of amnesia among ADAMs.

The following steps will allow you to enter and run the programs in this book:

1. Plug in your ADAM printer, memory module, and keyboard as shown in the manuals; hook the tv adapter to the VHF terminals on a tv.

2. Turn the power switch on the printer to on, and turn on the tv. When the printer is through making a few quick noises, the screen should display ADAM'S ELECTRONIC TYPEWRITER.

3. Open the tape drive door, insert the SmartBASIC datapack (label side out). Close the tape drive door.

4. Pull the reset button on the top of the memory module. The datapack should begin to whirr and click. In approximately 60 seconds, the whirring will stop, and the screen will display the SmartBASIC title screen:

```
        Coleco SmartBASIC Version 1.0
    ]
```

5. Type NEW.

6. Enter the program as it appears in the book. As you enter these programs, the ADAM will check each line for errors, placing a caret (^) under the spot where the first error is in the line.

EDITING PROGRAMS

To change a line that you have typed in use the following procedure:

1. List that line (for example, LIST 480 will put line 480 on the screen for you).

2. Move the cursor () to that line by holding down the control key while using the up/down arrow keys. Don't worry about moving over other words or lines to get there; it ignores those.

3. When you're on the line that you wish to change, release the control key, and use the left/right arrow keys to "copy" the parts of the line you want, erasing unwanted characters with the space bar. You can change characters by simply typing right over them. When you're done with that line, press the return key, and the new version of that line is placed in ADAM's memory.

SAVING PROGRAMS

When a program works the way you want it to, you can save it on a tape. A blank digital datapack was included with your computer. This is where you should store any programs that you write. The datapacks seem to have a large amount of storage available. A 10-line program, for example, takes up a single block on the tape, and the directory shows that you can get about 250 of these blocks onto a tape. It is not known by the authors if it is possible to store programs on the SmartBASIC tape itself, but it's best not to try until these details are released by either Coleco or a computer engineer familiar with the system. Remember, SmartBASIC is on the tape, and if you lose that, you lose the ability to do any programming.

The datapacks store your programs as a series of magnetic *ons* and *offs*, and are easily erased by strong magnetic fields. Common sources of magnetic fields are tv or monitor screens, printers, stereo speakers, and any appliances with large motors (washers, dryers, refrigerators, etc.). Keep your datapacks away from these things at all times.

The following steps will save a program onto a datapack:

1. Check your program to see if it runs correctly.

2. Type SAVE . . . . where . . . . is the name that you want to call the program. The 11 dots means that you can have 11 characters in the program name. When you look at the list of all of the program names (by typing CATALOG), you'll see that each name appears twice.

```
a 2 ORB.PERIODS
A 2 ORB.PERIODS
A 1 MASSCHANGE
a 1 MASSCHANGE
```

No, you're not seeing double. The programs with the A in front are the ones that you can load and save. The ones with the a in front are backup versions. If anything ever happens to the A version of MASSCHANGE, you can resurrect that program by typing RECOVER MASSCHANGE.

LOADING PROGRAMS

To copy a program from tape into memory, type LOAD

. . . . where . . . . is the name of the program. When the SmartBASIC prompt (]) appears on the screen, you can type RUN to begin the program on LIST to view the program lines.

ENTERING PROGRAMS

There are two ways to enter the programs that you want ADAM to run. You can type them in while SmartBASIC is in memory or you can use the SmartWRITER word processor. There are advantages and disadvantages to both.

Entering with SmartBASIC

The advantages of entering your program with SmartBASIC are as follows:

☐ Gives instant feedback on errors in basic.
☐ Allows you to run your program and watch its progress as you write it.
☐ Saves program storage space (BASIC programs take up four times as much room when stored with the word processor (there's a trick to get around this).
☐ Allows you to do calculations in BASIC's immediate mode.

The disadvantages of entering your program with SmartBASIC are listed here.

☐ Any lines with errors in them are clobbered. That is, BASIC deletes the line from the workspace. This only happens when entering lines: an error found while running the program is not fatal.
☐ Editing is done livescreen. There are no edit commands. The whole line must be copied using the arrow keys in order to change it. In addition, it is impossible to move the cursor to the leftmost column of the screen while editing. As this cursor movement is necessary to copy a clobbered line, you can see why clobbered lines can't be resurrected.
☐ Line listings in BASIC leave something to be desired. Any lines longer than 38 characters are outdented on their second and later screen lines. This is true of both the video and print outputs.
☐ The lock button on the ADAM keyboard is a shiftlock, not a caps-lock. While SmartBASIC will accept commands and keywords in both uppercase and lowercase, it needs to have all line numbers in lowercase. The shiftlock is a toggle; one press enables it, the next disables it, the next enables it, and so on. Shift does not disable it.

Entering with SmartWRITER

The advantages of using SmartWRITER are considerable.

☐ Editing is easier, as you have the full power of the word processor at your fingertips. While you still do not have edit commands, it is much easier to simply go to a part of the program that you're interested in and change, insert, or delete.
☐ Access is easier since you don't have to load SmartBASIC in order to program (SmartWRITER is in permanent ROM memory in ADAM).
☐ "What you see is what you get:" the word processor doesn't outdent program listings. You must, however, remember to end each line with a return (they look like a ▲ on the screen).
☐ Full feedback (sound) to help in typing.

The disadvantages of using SmartWRITER are not overpowering.

☐ No feedback of any errors in BASIC until the program is run. (No worse than most BASICs.)
☐ Large storage space needed by word processing files. This disadvantage can be overcome. Save programs as WP files (there's an H in front of them on the CATALOG. LOAD SmartBASIC, and LOAD the WP file into the BASIC workspace. Now SAVE the file with a slightly different name. DELETE the original file. Using this method, a 4-block WP file becomes a 1-block BASIC file.
☐ The shiftlock acts strangely: hitting backspace disables it, do keep your eye on the screen when editing.

RULES OF SYNTAX

Spaces seem to be very important. A seasoned Apple or TRS-80 programmer will probably start talking to him/herself over this one, but it never hurts to be careful. A FOR NEXT loop will not work if you forget spaces. SmartBASIC will not parse keywords and expressions for you. The need for spaces to identify keywords varies from function to function. For example, 10PRINT"HELLO will be parsed correctly, but 20FORX=1TO1000 will earn you the electronic raspberry and a Meaning Unclear Expecting ":" message. When in doubt, space. Luckily, SmartBASIC will use the caret (^) in its error message to tell you where it thinks the error is. It seems to only worry about the first error in a line with more than one error. Practice makes perfect.

NUMBERS ON THE SCREEN

Throughout the programs you will see numbers converted to a certain number of decimal places by using a statement like this:

```
X=INT (X*10^3+.5)/INT(10^3+.5)
```

This rounds numbers off to the number of places you desire, as long as the number of places that you desire is put where the "3" is in our example. This is needed because the ADAM's SmartBASIC sometimes does too much math and calculates out to 9 decimal places without being asked. Sometimes this can come in handy, but most of the time it's distracting. We've rounded off the places that you won't need.

SCREEN FORMATTING

One thing to keep in mind as you rush headlong into converting all known Applesoft programs to SmartBASIC is that the screen size is different. Apple IIs have 40 characters in two printing zones on their screens, while ADAMs have 31 in 2 zones. Typing in an Applesoft program verbatim will give you a mess on your screen if the PRINT statements try to print more than 31 characters per line.

REMARKS AND EXPLANATIONS

We have avoided the use of REM statements in the programs themselves. Instead, to explain the programs, there are line-by-line descriptions. These explanations of the programs will help you to identify a line or variable's job.

CATALOG PROGRAM

The catalog program can make life easier for you and your ADAM. It will automatically catalog the digital datapack each time you insert it. With your datapack in the drive, type in and run the program. Then type SAVE HELLO. The drive will whirr for a few seconds, and then display the] prompt. From now on, when you use this datapack, it will not display Coleco SmartBASIC V1.0, but will catalog your tape for you to show you what is on it.

Program Listing

```
10 D$ =  CHR$ (4)
20  REM  CTRL-D
30  PRINT D$;"CATALOG"
40  END
```


2

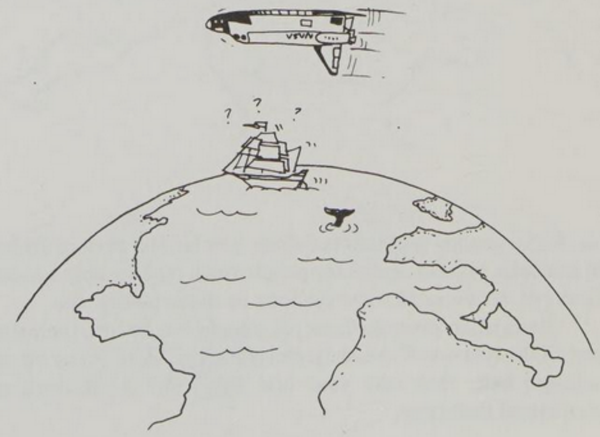
Adventures in Astronomy and Space Science

We start our adventures with a little jaunt covering over eight billion miles (round trip). On our way, we'll stop to explore local conditions, check on any satellites in the neighborhood, and give you a chance to live the life of an extraterrestrial all in the comfort of your own home.

Thanks to ADAM and some simple programs that we've supplied at the end of each section, you can simulate all of this activity without an army of engineers and a fleet of spaceships.

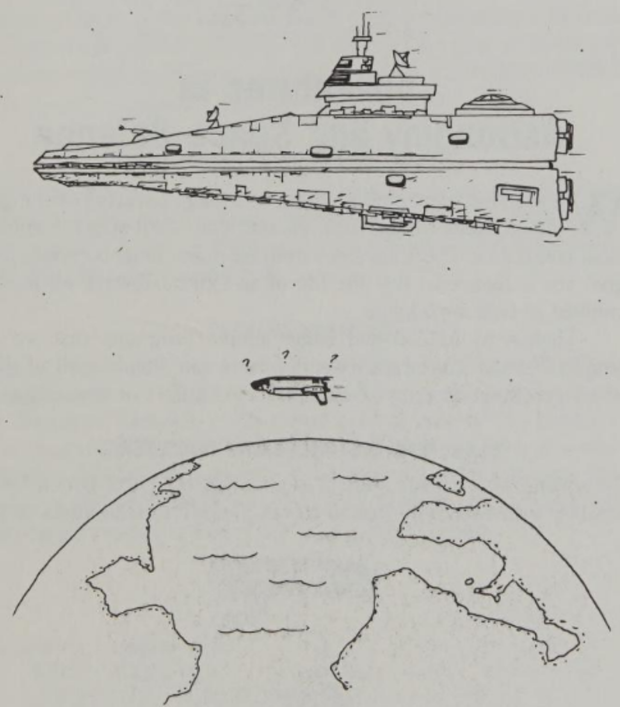
PLANETARY DISTANCES AND DIAMETERS

To anyone who travels a lot, it seems that the earth gets a little smaller with each new way to travel. It used to take years to go



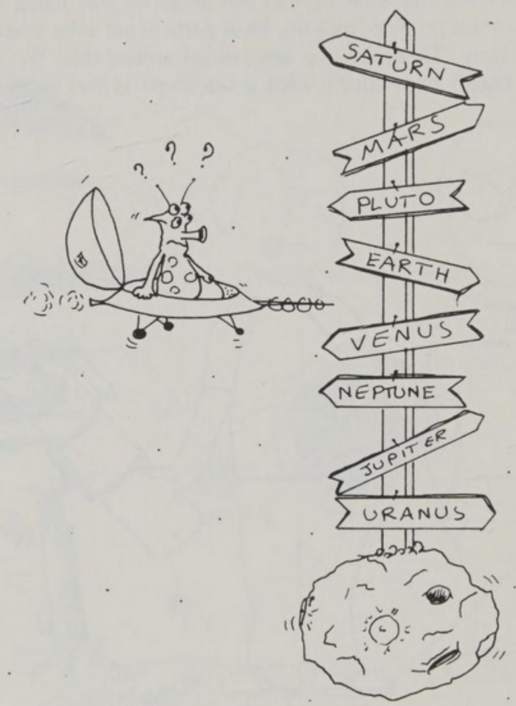
completely around the earth. Now you can do that trip in ninety minutes. As you can see, the difference isn't really because the earth has shrunk, but that the vehicles are getting faster. In fact, your average space shuttle passenger travels farther in one orbit than the most sidetracked sailor ever did.

For the time being, it seems that we'll have to wait for another big breakthrough (atomic spaceships or something like that) before humans will travel much faster than they do at present. Until then, the biggest thrill we'll get is from the new places we have to travel

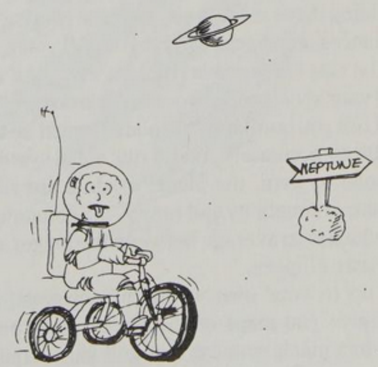


to. We've already sent robots to look at or land on most of the bodies in our solar system, and many people think that it's only a matter of time before we send some humans to those places, too.

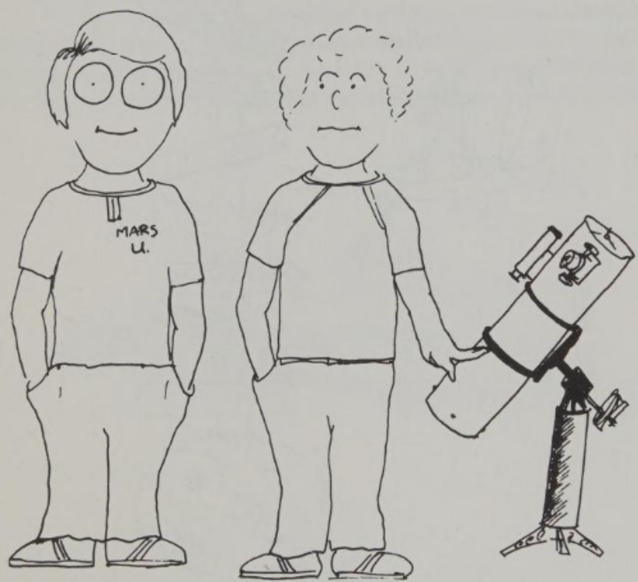
If and when we go to those places—Mars, Saturn, their moons, and all the rest—we'll need careful planning. "All the way out to the asteroid belt, then take your first left," just won't work at astronomical distances.



It would also be nice to know the sizes of the planets. Then we would be prepared for navigating the ones with land or oceans or atmospheres. Touring Neptune might turn out to be a tiring task if all you have is your trusty tricycle . . .



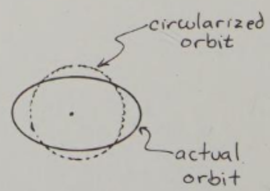
We can't see the solar system and judge its size using the equipment nature provided us with. Most parts of our solar system are too far away. There are two ways to get around this. We can either get bigger eyes (that's what a telescope is for) or get a



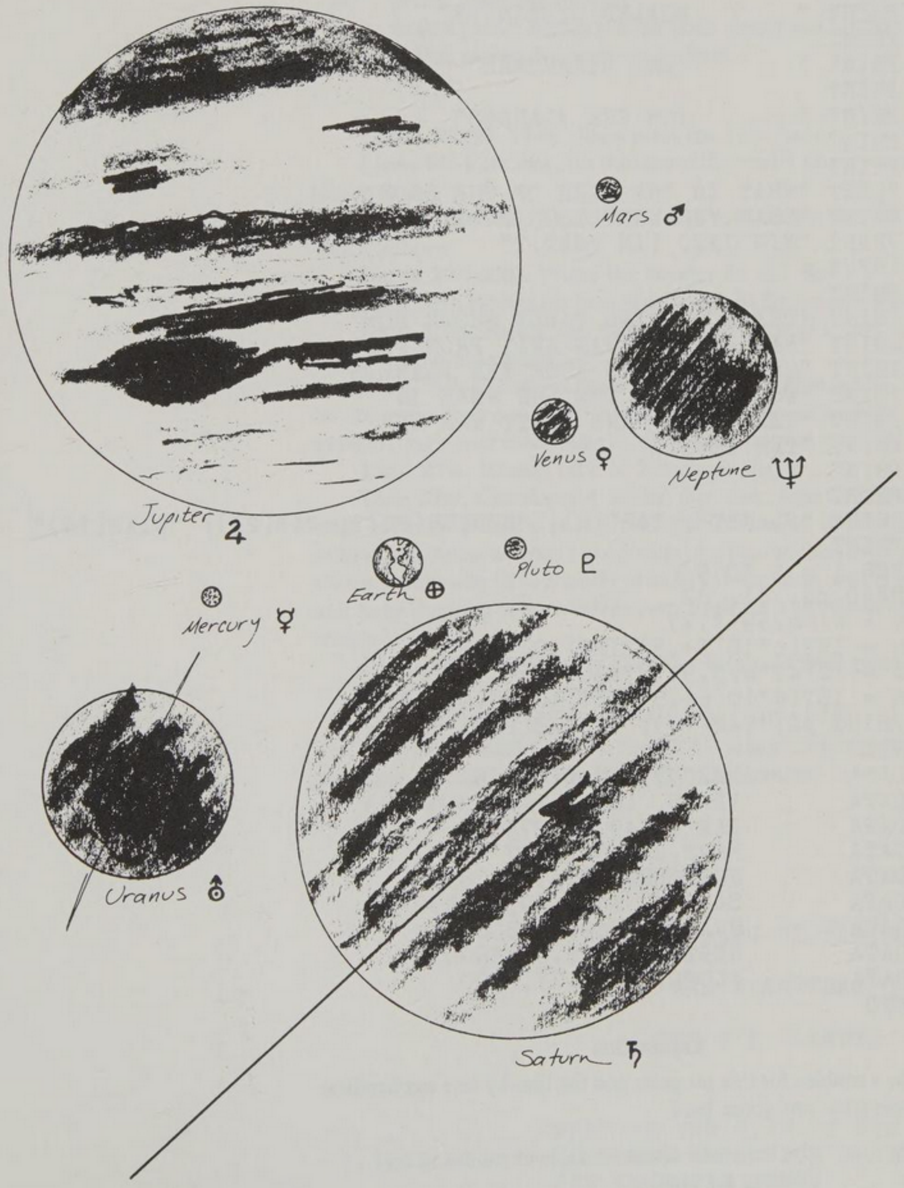
smaller solar system. Making your own small solar system is easy, once you know how to scale things. For instance, the sun's diameter is roughly 100 times as large as the earth's. This means that if you use a baseball as your earth model you will need a 27-foot-diameter sphere to be used as the sun. Such objects aren't normally found lying around your average home or school.

To make building these models simpler, the program will do all the scaling of distances and diameters for you. All you need to do is tell the program the size of the space that you want your model to fit into (the length of your shoebox, bedroom, classroom, gym, football field, etc.). It will tell you how large the planets must be and how far apart they must be to accurately model our solar system.

In the real solar system, the planet's orbits are ellipses, not circles. For the sake of simplicity and sanity, the computer will give you an orbit size which is an average between the longer and shorter diameters of the true ellipses.



The rest is up to your own wonderful imagination . . . an encyclopedia can give you maps of the planet surfaces for paper mache globes, or in a pinch, you can use golf balls, frisbees, tires . . . Bon Voyage!



Program Listing

```
10 HOME
20 PRINT "        SCALED DISTANCES"
30 PRINT
40 PRINT "         AND DIAMETERS"
50 PRINT
60 PRINT "        FOR THE PLANETS"
70 PRINT
80 PRINT
90 PRINT "WHAT IS THE SIZE OF THE ROOM"
100 PRINT "THAT YOU WANT YOUR MODEL TO"
110 PRINT "FIT INTO (IN FEET)?"
120 INPUT m
130 PRINT
140 PRINT "THE FOLLOWING TABLE SHOWS HOW"
150 PRINT "BIG AND HOW FAR AWAY FROM THE"
160 PRINT "CENTER OF THE ROOM THE PLANETS"
170 PRINT "WOULD BE IF THE SUN WERE IN"
180 PRINT "THE CENTER AND PLUTO WAS ON"
190 PRINT "THE EDGE."
200 PRINT
210 PRINT
220 PRINT "PLANET"; TAB(10); "CENTER(FT)"; TAB(22); "DIAM(IN)"
230 PRINT
240 FOR i = 1 TO 9
250 READ p$, d1, d2
260 c = d1*m/5911.77
270 c = INT(c*10^3+.5)/INT(10^3+.5)
280 d = 12*d2*m/5.91177E+09
290 d = INT(d*10^6+.5)/INT(10^6+.5)
300 PRINT p$; TAB(10); c; TAB(22); d
310 NEXT i
320 DATA       MERCURY,57.91,4868
330 DATA       VENUS, 108.2,12112
340 DATA       EARTH,149.6,12756
350 DATA       MARS,227.94,6787
360 DATA       JUPITER,778.33,142830
370 DATA       SATURN, 1429.99,119330
380 DATA       URANUS,2869.57,50800
390 DATA       NEPTUNE,4496.6,48600
400 DATA       PLUTO, 5911.77,2900
410 END
```

Explanation

The variables for this program and the line-by-line explanation of its operation are given here.

m	the expected diameter of your model in feet
i	counter for planets
p$	planet name
d1	average planet distance from sun (in millions of kilometers)
d2	planet diameter (in km)
c	scaled planet distance from your model sun (in feet)
d	scaled planet diameter (in inches)

Lines 10-80. These lines print the introduction.

Lines 90-120. Asks for the size of the model which you would like to make.

Lines 130-210. Introduction and explanation for the chart which follows.

Lines 220-230. Prints the Header for the chart.

Line 240. Begins the loop of calculations for the nine planets.

Line 250. Reads the name(p$), distance(d1), and diameter(d2) of the nine planets.

Line 260. Calculates the scaled distance for the model, based on the ratio between your model size and the size of the solar system (times your model size m, divided by the real size 5911.77.

Line 270. Rounds off c to 3 decimal places.

Line 280. Calculates d, scaled diameter, based again on the ratio between scaled size and real size (you need the magnitude million here because you're not converting like-units as in line 260), then multiplies by 12 to convert to inches (believe us, feet is not the unit to be thinking of here unless your model is a few thousand feet wide).

Line 290. Rounds your diameter to six decimal places.

Line 300. Prints the results.

Line 310. The NEXT planet for line 240's FOR.

Lines 320-400. Data for the READ in line 250; the elements are name, distance in millions of km, and diameter in km.

Line 410. The end.

Sample Run

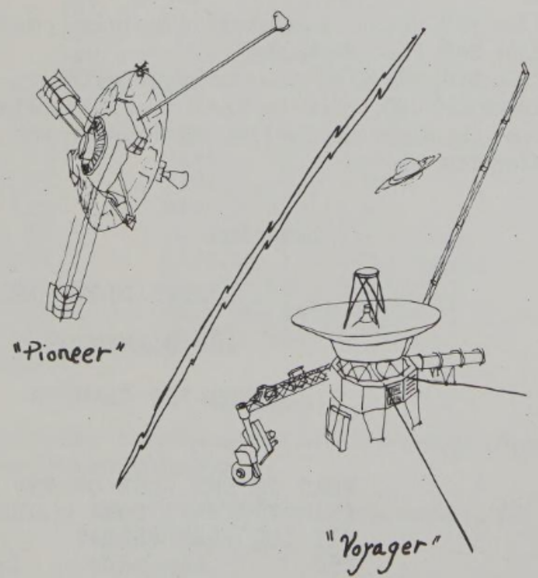
```
        SCALED DISTANCES

         AND DIAMETERS

        FOR THE PLANETS


WHAT IS THE SIZE OF THE ROOM
THAT YOU WANT YOUR MODEL TO
FIT INTO (IN FEET)?
?50
```

THE FOLLOWING TABLE SHOWS HOW BIG AND HOW FAR AWAY FROM THE CENTER OF THE ROOM THE PLANETS WOULD BE IF THE SUN WERE IN THE CENTER AND PLUTO WAS ON THE EDGE.

PLANET	CENTER(FT)	DIAM(IN)
MERCURY	.49	4.94E-04
VENUS	.914999999	1.229E-03
EARTH	1.265	1.295E-03
MARS	1.928	6.89E-04
JUPITER	6.583	.014496
SATURN	12.094	.012111
URANUS	24.27	5.156E-03
NEPTUNE	38.031	4.933E-03
PLUTO	50	2.94E-04

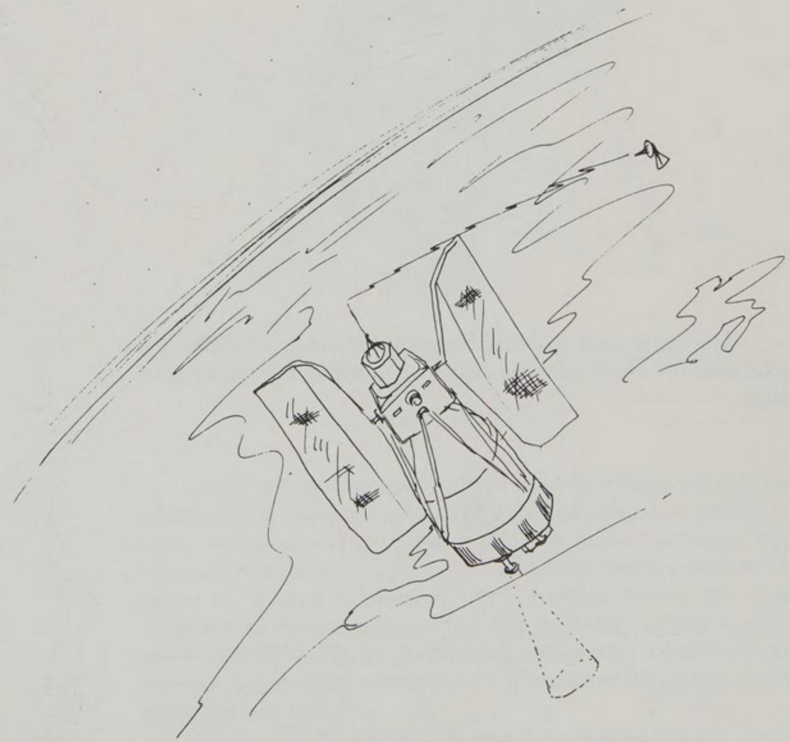
SATELLITE ORBITAL PERIODS

When you watch science fiction shows, you notice that the commanders usually send out *probes* before they get too close to the planet they're approaching. This is actually a pretty smart move. Better you should lose a probe to the mercy of a Rigelian Cloud Snake (ugh!) than your First Officer and Ship's Doctor.

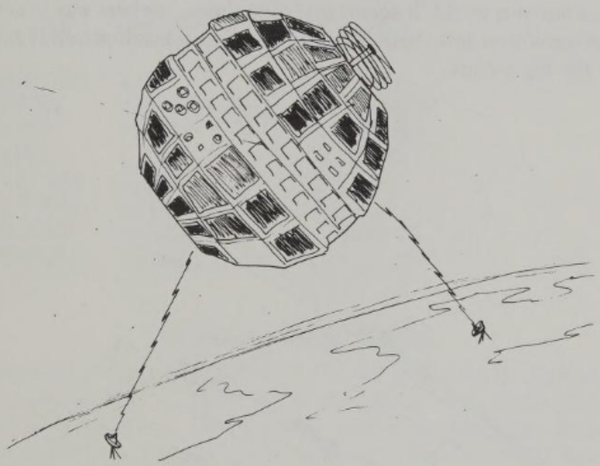


A probe is really just a highly specialized robot. We humans use them, and we call them satellites if they orbit the earth (like Sputnik or Telstar), or spacecraft if they travel to another planet (like Pioneer or Voyager).

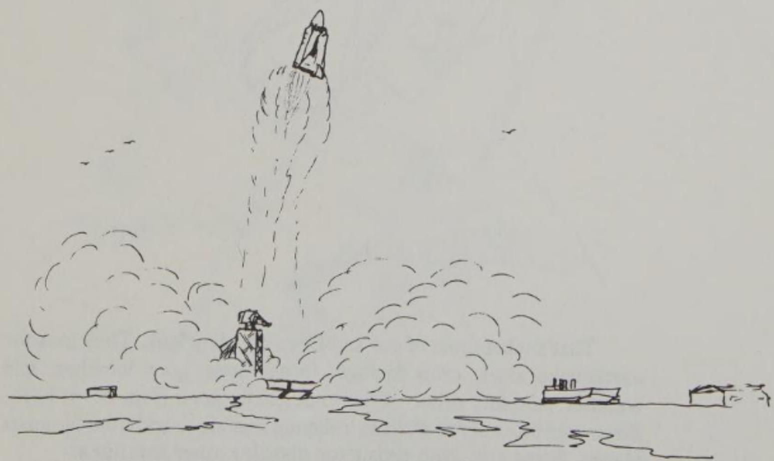
The satellites, or earth-orbiting probes, are designed to gather information, not about any extraterrestrials or distant worlds, but about our own world. It seems that sometimes, the best way to see what's going on down here is to step back (a few hundred miles) and get the big picture.



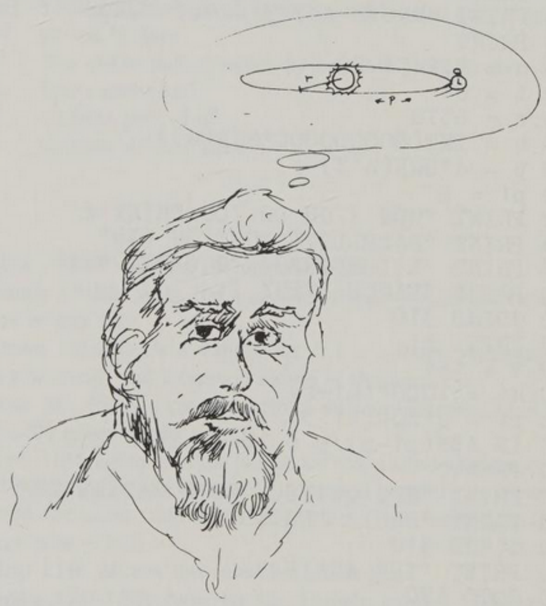
That's what most of our satellites do, they look. They look for earthquake sites, crop damage, forest fires, good weather, bad weather, and any other objects not easily seen from down here. Some satellites serve as ears, relaying signals to and from far apart places. Some can even repair (or disable) other satellites.



How do they get up there? We put them on top of a controlled explosion (a rocket) strong enough to get them as high as we need them.



How do we keep them up there? As long as the satellite is far enough away and has a certain amount of energy, it will counteract the pull of the earth's gravity. How much energy? See the adventure on escape velocity for those details. How far away?



Johannes Kepler

Johannes Kepler figured this out years ago. He found that the square of the planet's (or satellite's) period (period × period) is equal to the cube of its distance from the mother planet or star (distance × distance × distance). Period is the time it takes for the planet or satellite to complete one revolution (orbit). Now that we've gotten these complex actions into mathematical form, it's fair game for ADAM. The programs can go either way; if you know the distance, you can find out the period. If you know the period, it will figure the distance.

And if you'd like a little challenge, the program will give you a period or distance and ask you to guess (or calculate) the unknown.

Some special applications of this program would be to design custom orbits for satellites (or space stations) that need to be *geosynchronous* (over a constant point on the surface) or super slow or super fast.

Happy landings!

Program Listing

```
 10 HOME
 20 PRINT "      COMPUTING SATELLITE"
 30 PRINT "        ORBITAL PERIODS"
 40 PRINT
 45 PRINT "ENTER ANY 3 DIGIT NUMBER": INPUT s: s = 0-s
 50 PRINT
 60 c = 5.245E-09
 70 t = 0
 80 r = 6378
 90 h = INT(800000000*RND(s))
100 p = c*SQR(h^3)
110 p1 = p
120 PRINT "HOW HIGH DO YOU THINK A"
130 PRINT "SATELLITE MUST BE (IN"
140 PRINT "KILOMETERS) TO ORBIT THE"
150 PRINT "EARTH EVERY ";
160 GOSUB 310
170 INPUT h1
180 t = t+1
190 h1 = 1000*(h1+r)
200 p1 = c*SQR(h1^3)
210 IF ABS(p1-p)/p < 1E-03 THEN 280
220 PRINT
230 PRINT "NOT QUITE.  YOUR SATELLITE"
240 PRINT "HAS A PERIOD OF "
250 GOSUB 310
260 PRINT "TRY AGAIN."
270 GOTO 170
280 PRINT "****   GOOD FOR YOU!   ****"
290 PRINT "YOU GOT IT IN "; t; " TRIES."
300 END
310 p4 = INT(p1/1440)
320 p5 = INT(p1-1440*p4)
330 IF p4 = 0 THEN 350
340 PRINT p4; " DAYS, ";
350 p2 = INT(p5/60)
360 p3 = INT(p5-60*p2+.5)
370 PRINT p2; " HOURS, ";
380 PRINT p3; " MINUTES?"
390 RETURN
```

Explanation

The variables and the line-by-line explanations are as follows:

c	constant to convert from astronomical units and years to meters and days/hours/minutes
t	number of tries

r	radius of earth in kilometers
h	randomly selected altitude for the satellite, between zero and 8 million meters
p	period of the satellite picked by computer
p1	period of satellite picked by computer
h1	your guess for the altitude needed for that period (p1)
p4	period in days
p5	part of the period which is less than 1 whole day
p2	hours part of p5
p3	minutes part of p5
s	random number seed

Lines 10-40. Prints the introduction.

Lines 45-50. Asks for a number used to generate random numbers in line 90.

Lines 60-80. Sets values for c (conversion constant), t (number of tries), and r (metric radius of the earth).

Line 90. Picks a random orbital altitude between zero and 800,000,000 kilometers.

Line 100. Computes the period for that altitude based on p^2 = h^3 (multiplied by the conversion constant for unit); to solve for p, you take the square root of the other side of the equation, so the p = square root of h^3.

Line 110. Moves the value for p into p1.

Lines 120-150. Asks for the altitude needed to keep the satellite orbiting with the following period.

Line 160. Goes to the subroutine (lines 310-390) that converts the period to days, hours, and minutes.

Line 170. Accepts your guess for the altitude.

Line 180. Adds 1 to your try count.

Line 190. Adds your guess to the radius of the earth, and converts to meters by multiplying by 1000.

Line 200. Computes the period for your guess, using the same method as line 100.

Line 210. Calculates the fraction of your guess that is in error. If this is less than .001, it means you've come close enough; if it is greater, it has you try again.

Lines 220-240. Announces that your error is too large.

Line 250. Goes to the subroutine which turns your period into days, hours, and minutes.

Lines 260-270. Asks you to try again and goes to the input at line 170.

Lines 280-300. Message displayed if your error is small enough and then ends the program.

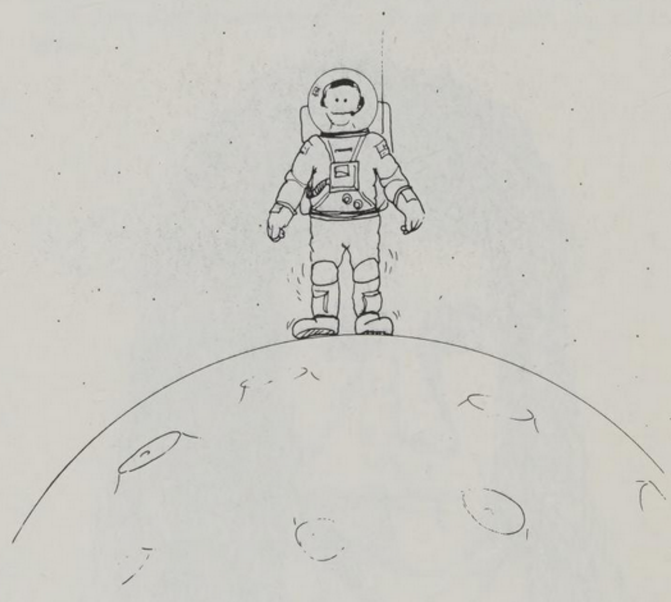
Lines 310-390. Subroutine to convert the period from minutes to days, hours, and minutes. 1440 = minutes per day. If there are less than 1440 minutes, it won't print any days; p5 is the leftovers from whole days; p2 divides this by 60 to find the number of hours; what's left over from that is divided by 60 to find p3, the number of minutes. Lines 340, 370, and 380 print these and then return to the line where they left off.

Sample Run

```
        COMPUTING SATELLITE
         ORBITAL PERIODS

ENTER ANY 3 DIGIT NUMBER
?123

HOW HIGH DO YOU THINK A
SATELLITE MUST BE (IN
KILOMETERS) TO ORBIT THE
EARTH EVERY 72 DAYS, 3 HOURS, 16 MINUTES?
?700000

NOT QUITE.  YOUR SATELLITE
HAS A PERIOD OF
68 DAYS, 9 HOURS, 9 MINUTES?
TRY AGAIN.
?750000

NOT QUITE.  YOUR SATELLITE
HAS A PERIOD OF
75 DAYS, 18 HOURS, 27 MINUTES?
TRY AGAIN.
?725000

NOT QUITE.  YOUR SATELLITE
HAS A PERIOD OF
72 DAYS, 1 HOURS, 2 MINUTES?
TRY AGAIN.
?726000
****   GOOD FOR YOU!   ****
YOU GOT IT IN 4 TRIES.
```

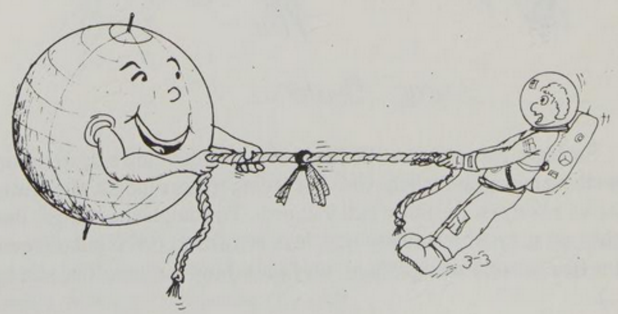

WEIGHT ON DIFFERENT PLANETS

It's pretty easy for most of us to get around on earth. That makes sense: our bodies have evolved and developed in the earth's gravitational field, and we would expect to be equipped with legs that can support our body.



Adventurers that we are, let's leave the earth and see what happens elsewhere. How will our legs do if we change the ground rules (no pun intended . . .). Before we go, let's see what the experts have to say about what we should expect.

Gravity is a form of energy which causes a force to be exerted



by an object upon another object. It attracts the objects to each other. We don't yet know what carries this force or exactly how it works, (now there's a problem to solve), but we know that it's there. If you have any doubts, try getting away from the earth and staying that way. (The trying is known as jumping, the proof of gravity known as falling. Please don't try this for too long, it tends to be painful.)

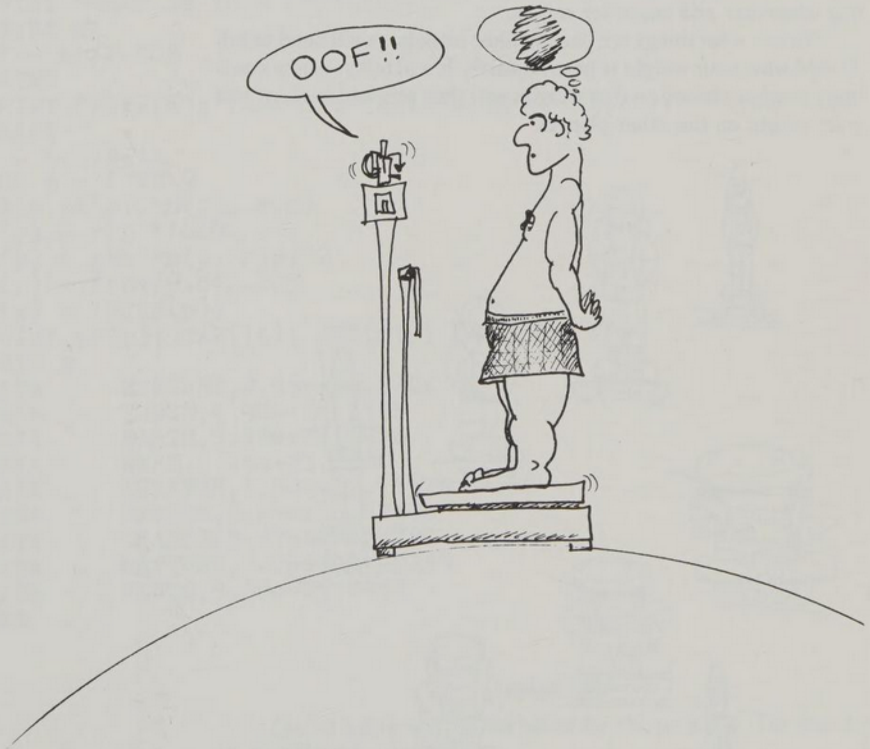
Sir Isaac Newton was the first person to develop a system for describing how gravity behaves. According to his reports, the more massive a body is, the more pull it exerts. To complicate things, the further away two obejcts are, the less attraction there is between them (that's comforting when you figure how massive the sun is . . .).

All of this can be written mathematically in a single equation that looks like the one in the margin.

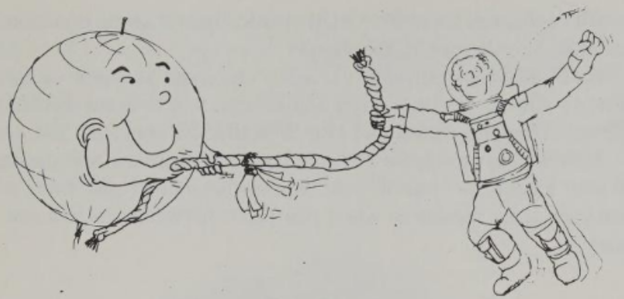
$$F = \frac{g \times m}{r^2}$$

As in all of our adventures, when we come across a useful equation, you can be sure that it will be somewhere in our BASIC program. This one, translated into SmartBASIC is in line 150.

Now we have a way for you and ADAM to figure out how much pull your local planet has on you. We call this pull weight, and it's what your scale measures when you put it between you and the planet.

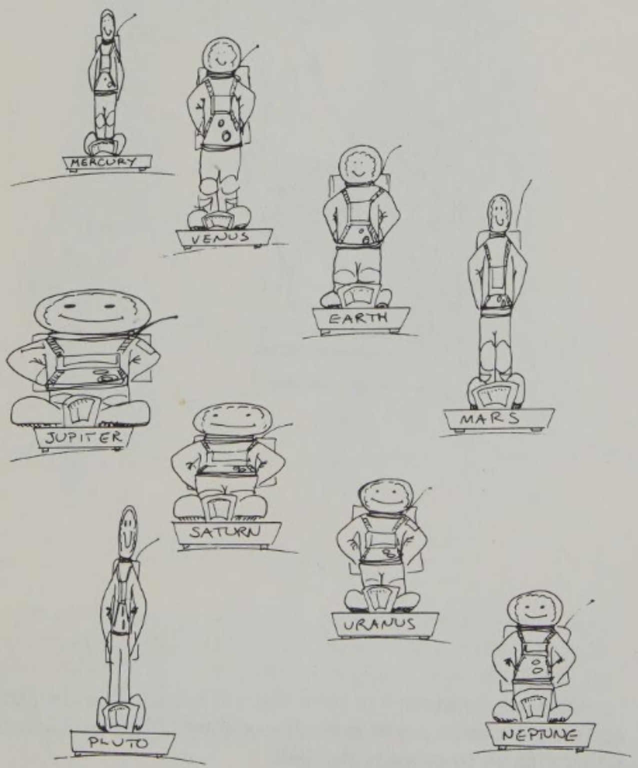


You may be pleased to know that you're pulling on the planet earth too, but since you're so much less massive than the earth, it's rather difficult to measure that pull.



So far, so good. Now how about that adventure we promised you? First let's get ADAM ready. As usual, type in the program for this adventure and check for errors.

To see what things are like on other planets, you'll need to tell ADAM what your weight is here on earth. It will tell you how much mass you have based on that weight, and then proceed to show you your weight on the other planets.



We did a little bit of homework for you and have included in the program the names and sizes (mass and radius) of the nine planets in our solar system. If you'd like, you can tell ADAM your own numbers, and see what it would be like on an asteroid, a particular moon, the sun itself (yikes!), a white dwarf star, or a neutron star. You can check the local library for the mass and radius of all of these astronomical oddities. Happy landings!

Program Listing

```
 10 HOME
 20 PRINT " WEIGHTS ON DIFFERENT PLANETS"
 30 PRINT
 40 PRINT
 50 PRINT "WHAT IS YOUR WEIGHT?"
 60 INPUT m1
 70 m1 = m1/2.205
 80 PRINT
 90 PRINT "PLANET"; TAB(10); "WEIGHT-KG"; TAB(22); "WEIGHT-LBS"
100 PRINT
110 g = 6.67E-11
120 FOR p = 1 TO 9
130 READ p$(p), m(p), r(p)
140 r(p) = r(p)*1000
150 f(p) = g*m1*m(p)/r(p)^2
160 f(p) = f(p)/9.8*2.205
170 f(p) = INT(f(p))
180 PRINT p$(p); TAB(14); INT(m1); TAB(27); f(p)
190 NEXT p
200 DATA      MERCURY,2.99e+23,2424
210 DATA      VENUS,4.90e+24,6128
220 DATA      EARTH,5.98e+24,6378
230 DATA      MARS,6.46e+23,3380
240 DATA      JUPITER,1.90e+27,71370
250 DATA      SATURN,5.69e+26,60400
260 DATA      URANUS,8.67e+25,23790
270 DATA      NEPTUNE,1.05e+26,22259
280 DATA      PLUTO,5.98e+23,2998
290 END
```

Explanation

Listed below are the variables for the program. The line-by-line explanation follows.

m1	your weight converted to kilograms of mass universal
g	gravitational constant
p	counter for the planets
p$	planet name

m planet's mass (in kg)
r planet's radius (in km)
f force exerted on your body due to gravity on the planet (your weight)

Lines 10-40. Prints the introduction.

Lines 50-70. Asks for your earth weight in pounds, then converts this to mass in kilograms (at 2.205 pounds per kilogram).

Lines 80-100. Prints Header for chart of mass and weights.

Line 110. Sets g = .0000000000667, the universal gravitational constant.

Line 120. Begins the loop to do calculations for all nine planets.

Line 130. Reads the data from lines 200-280. The first element is the name, the second is the mass, and the third is the radius.

Line 140. Converts radius to actual value. You could skip this step and add three zeros to each radius in the data statements, but doing it as a step saves space.

Line 150. The SmartBASIC version of the equation which you saw in the text (force equals gravitational constant times your mass times the planet's mass, all divided by the square of the radius).

Line 160. Converts this force (in newtons) to pounds (at 1/9.8 kilograms per newton and 2.205 pounds per kilogram).

Line 170. Turns f into an integer. Once you see how large these can be, the decimal places become unimportant.

Line 180. Prints the planet name, your mass, and your weight on that planet.

Line 190. NEXT planet for line 120's FOR.

Lines 200-280. Data for the planets (name, mass, and radius in thousands of km).

Line 290. The end.

Sample Run

```
 WEIGHTS ON DIFFERENT PLANETS


WHAT IS YOUR WEIGHT?
?110
```

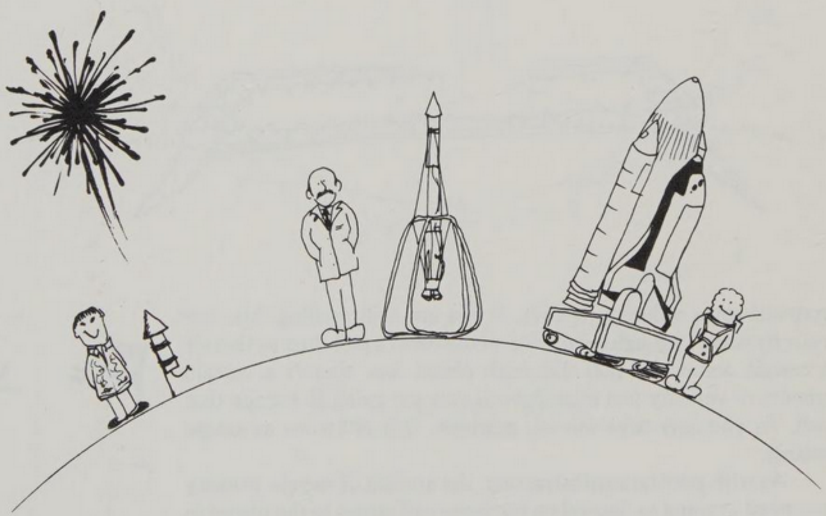
PLANET	WEIGHT-KG	WEIGHT-LBS
MERCURY	49	38
VENUS	49	97
EARTH	49	110
MARS	49	42

PLANET	WEIGHT-KG	WEIGHT-LBS
JUPITER	49	279
SATURN	49	116
URANUS	49	114
NEPTUNE	49	158
PLUTO	49	49

ESCAPE VELOCITY

This adventure deals with escaping from the gravitational field (pull) of a large massive body (planet). Naturally, if our last adventure left you on some strange planet checking your weight, the least we can do is show you how to get off that planet. We don't want any of you to be late for supper.

The usual way to get off a planet is with a rocket. Depending on how far away you want to get, you use a different type of rocket.

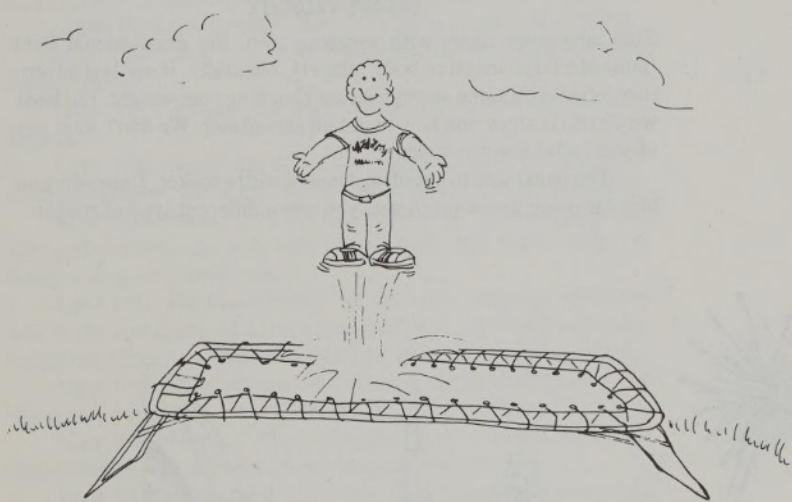


The Chinese wanted to propel their fireworks into the air for festivals, so they used small, inexpensive ones. Robert Goddard simply wanted to prove that rockets could be safely launched and recovered, so his were no bigger than necessary to get themselves off the ground.

Nowadays, we need to use rockets to put other objects

(payloads) into orbit and keep them there. These rockets, therefore, must be designed to quickly lift far more than their own weight.

In the last adventure, you tried to escape from the earth's pull by jumping. You contracted the muscles in your legs, and gave



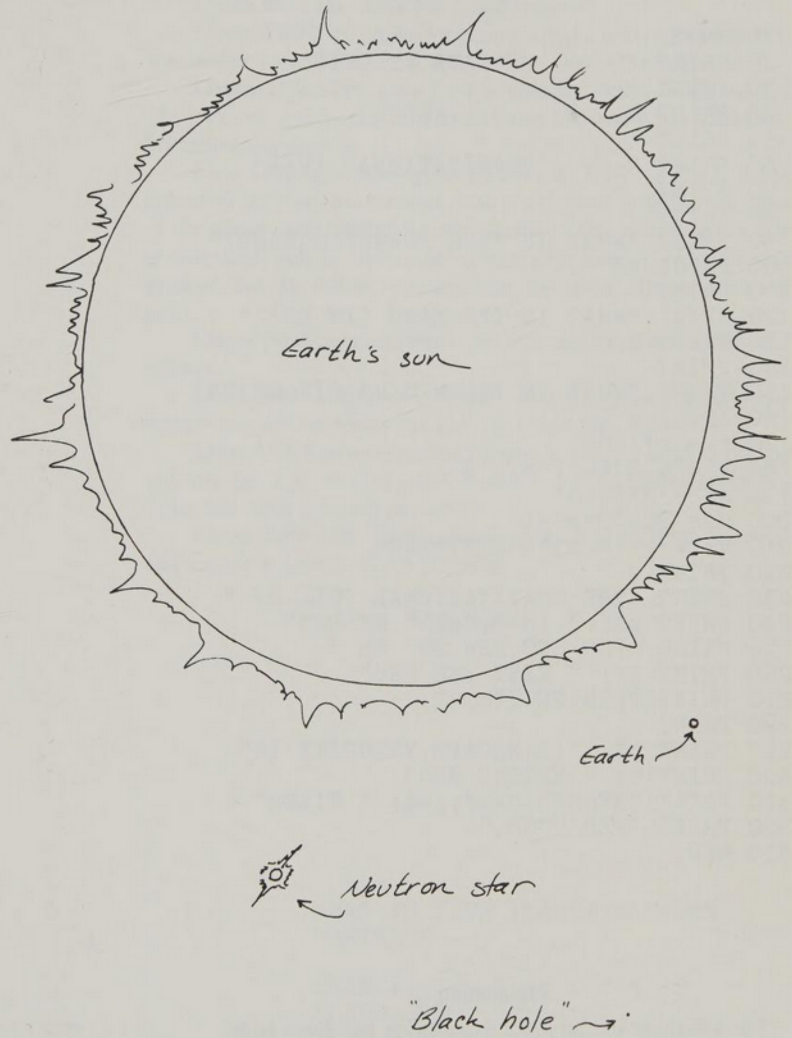
yourself some velocity (speed). If you are still reading this, the velocity wasn't enough to counteract the earth's pull. Just as there's a certain amount of pull that each planet has, there's a certain amount of velocity you must have if you are going to escape that pull. As you may have already guessed, this is known as *escape velocity*.

$$F = \frac{g \times m}{r^2}$$

As with gravitational attraction, the amount of escape velocity you need is going to depend on the mass and radius of the planet in question. The first equation which the program uses looks like one in our last program.

It will figure the pull exerted by a planet. The g that results is used in a second equation. This equation (line 100 in the program) is the one which figures the escape velocity. It too, depends on the radius of the planet. What is interesting here is that you keep the same mass in several examples, and only change the radius. This is what physically happens when a star becomes a white drawf, then a neutron star, and finally a black hole. A black hole is defined, by the

$$V_e = \sqrt{2 \times g \times r}$$



way, as a star whose escape velocity is higher than the speed of light. That's why it's black—not even light can get away from it to reach your eyes. You may want to figure out, by changing radio, how small the earth would have to become in order to be a black hole (never fear: since the earth isn't a star, this will never actually happen).

Program Listing

```
10 HOME
20 PRINT "       ESCAPE VELOCITY"
30 PRINT
40 PRINT "            AND"
50 PRINT
60 PRINT "      GRAVITATIONAL PULL"
70 PRINT
80 PRINT
90 PRINT "WHAT IS YOUR PLANET'S NAME?"
100 INPUT p$
110 PRINT
120 PRINT "WHAT IS ITS MASS (IN KG)?"
130 INPUT m
140 PRINT
150 PRINT "WHAT IS ITS RADIUS (IN KM)?"
160 INPUT r
170 r = r*1000
180 g = 6.67E-11*m/r^2
190 gf = g*3.281
200 v = SQR(2*g*r)
210 vm = v*1E-03*.62137*3600
220 PRINT
230 PRINT "THE GRAVITATIONAL PULL OF "
240 PRINT p$; " is "; g; " METERS"
250 PRINT "PER SEC PER SEC OR "
260 PRINT gf; " FEET PER SEC"
270 PRINT "PER SEC."
280 PRINT
290 PRINT p$; "'S ESCAPE VELOCITY IS"
300 PRINT v; " METERS PER"
310 PRINT "SECOND OR "; vm; " MILES"
320 PRINT "PER HOUR."
330 END
```

Explanation

The variables and program explanation are given here.

p$	the planet's name
m	the planet's mass (in kilograms)
r	the planet's radius (in kilometers)
g	gravitational pull (in meters/sec/sec)
gf	gravitational pull (in feet/sec/sec)
v	escape velocity in meters per second
vm	escape velocity in miles per second

Lines 10-80. Prints the introduction.

Lines 90-110. Asks for planet's name.

Lines 120-140. Asks for planet's mass in kilograms (6 trillion metric tons would be 6,000,000,000,000,000 kilograms!).

Lines 150-170. Asks for the planet's radius in kilometers (the way you usually find it in books), line 170 converts radius from kilometers to meters.

Line 180. Calculates gravitational pull: 6.67 × 10^−11 is the universal gravitational constant, which when multiplied by the mass of the planet and divided by the square of the radius, gives you gravitational pull (notice as the planet gets heavier, the pull gets greater, but the further you are from the planet, the smaller the pull).

Line 190. Converts metric pull to English pull at 3.28 feet per meters.

Line 200. Calculates escape velocity, which equals the square root of two times the pull (g) times the radius.

Line 210. Converts metric escape velocity to English escape velocity (at .001 meters per kilometer; 0.62137 kilometers per mile; and 3600 seconds per hour).

Lines 220-330. Displays gravitational pulls (both formats) and escape velocities (both formats).

Sample Run

```
       ESCAPE VELOCITY

            AND

      GRAVITATIONAL PULL


WHAT IS YOUR PLANET'S NAME?
?GRINDUK

WHAT IS ITS MASS (IN KG)?
?698000000

WHAT IS ITS RADIUS (IN KM)?
?34500

THE GRAVITATIONAL PULL OF
GRINDUK is 3.91149758E-17 METERS
PER SEC PER SEC OR
1.28336235E-16 FEET PER SEC
PER SEC.
```

GRINDUK'S ESCAPE VELOCITY IS
5.19512592E-05 METERS PER
SECOND OR 1.16211434E-04 MILES
PER HOUR.

DISTANCE TO THE SUN

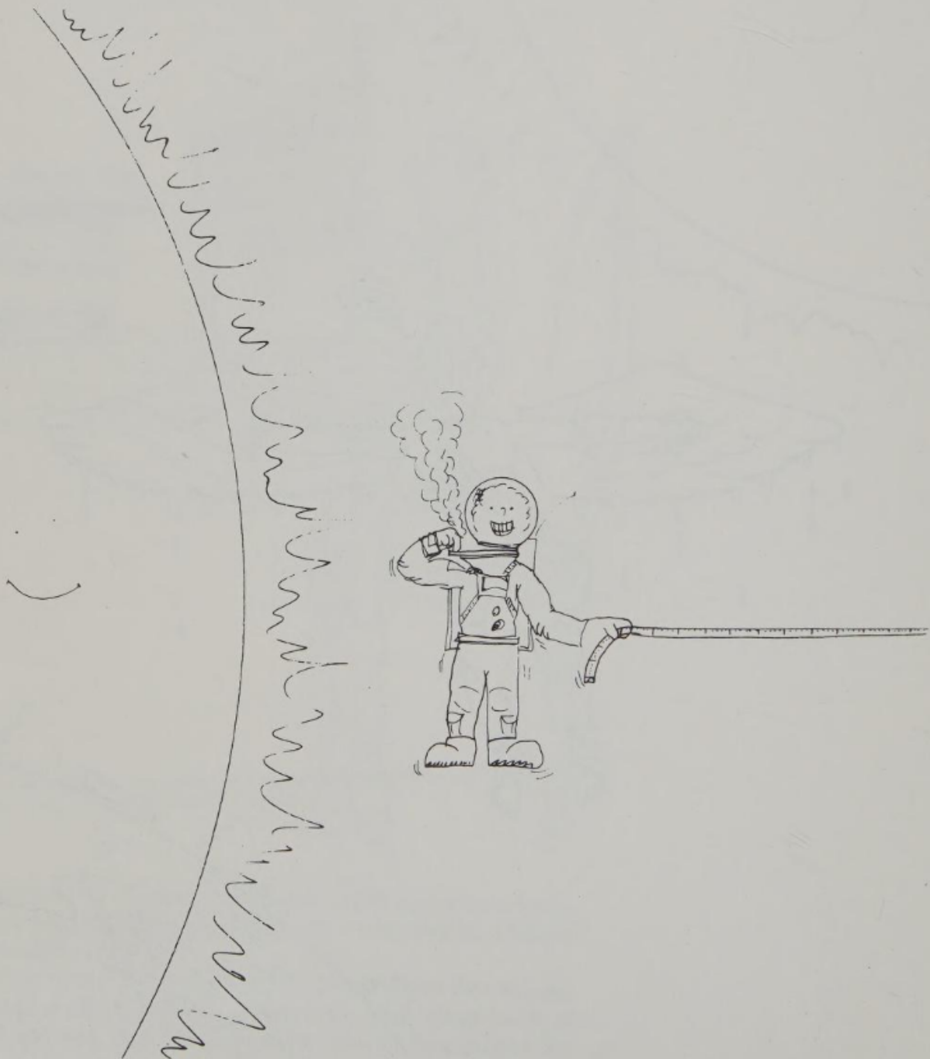
To be perfectly honest, the easiest way to find the distance from the earth to the sun is to look it up in the encyclopedia.

But it's hardly the best way. And we didn't always have books to rely on. Somebody had to be the first one to calculate the distance from the earth to the sun.

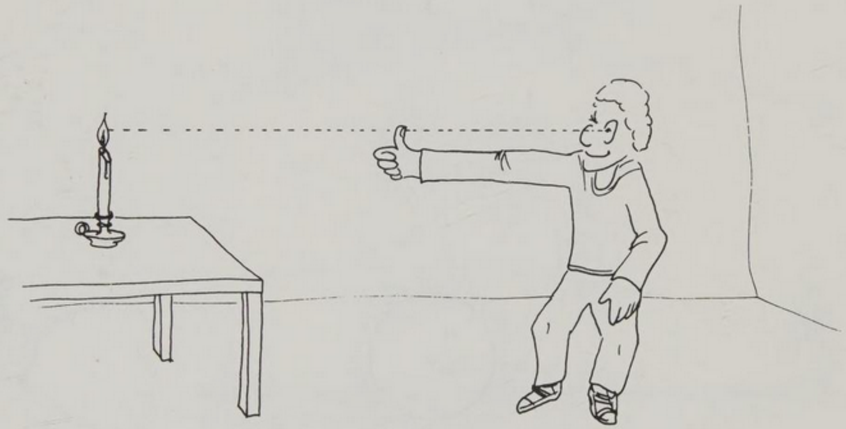
People have not always had a good idea of where the sun was. According to Greek mythology, Daedelus, with wings made of feathers and wax, flew too close to the sun. Its heat melted the

wings. This story was perfectly acceptable to people who were estimating the distance to the sun based on the image they saw and the heat they felt. Life isn't quite that simple. What was needed was a method of méasurement.

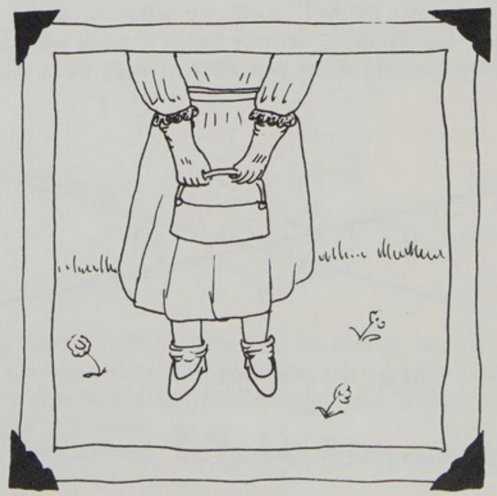
There's one problem: direct measurement is tricky when you're dealing with the sun. How would you like to be the one holding the far end of the tape measure?



Let's try an *indirect* method of measurement. There are a few of these to pick from. One of the methods involves the use of ***parallax***, the apparent change in position of a faraway object when the viewpoint is changed. You can see the effect of parallax by closing one eye, lining up an outstretched hand with a distant object, and the alternating closing one eye and then the other. See the effect?



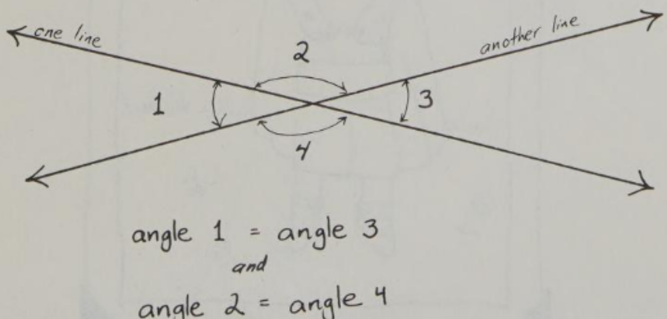
Parallax is also responsible for all of those snapshots of Aunt Gertie's knees, when you were sure that you had her lined up just right in the viewfinder (the viewfinder and lens on some cameras have different viewpoints).



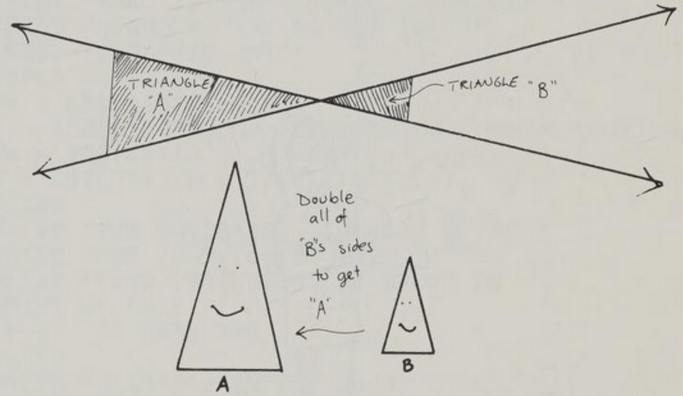
When astronomers measure the distance to the stars with this method, they don't blink. Instead, they put their eyes (telescopes) at viewpoints as far apart as possible (two views, six months apart, as the earth orbits the sun) and measure the apparent shift. From this, they can calculate the distance which the shift represents. The

laws of trigonometry (angle measurement) dictate the relationship between angles and distance.

The simpler method, which we will use, draws on trigonometry, too. To use it, we rely on the fact that when two lines cross, they can only create two different angle sizes. The angles

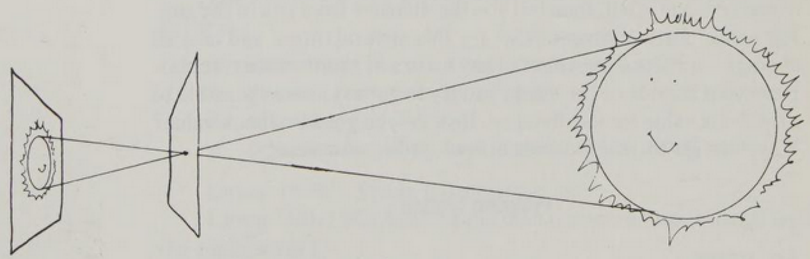


which lie across from each other will always have the same measure. If you make triangles out of two of these angles, the triangles will be similar. This means that even if their sizes are not the same, the relative lengths of the sides stays the same. If you multiply the



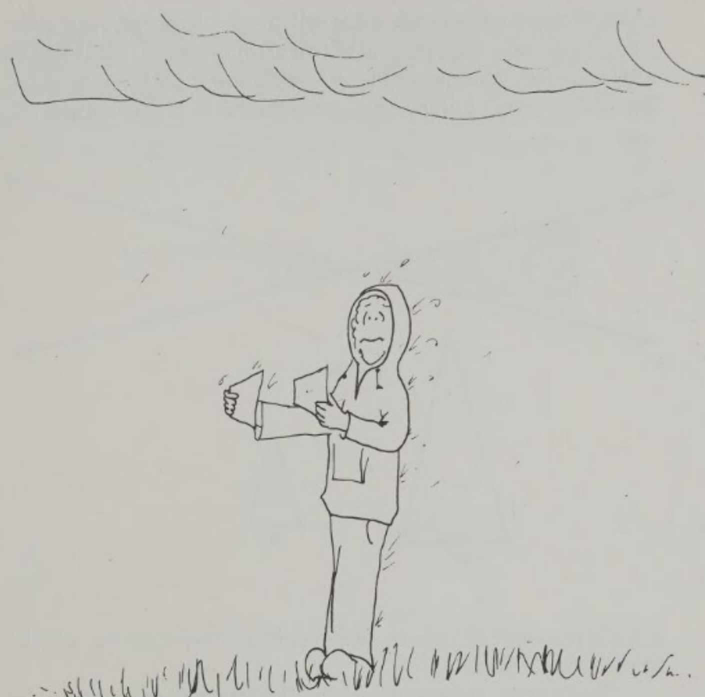
sides of the small triangle by some number, you can get the lengths of the large triangle.

That's what we're going to do. The triangles are going to be made using a pinhole (in an index card) as the point of intersection. The far edge of the large triangle will be the diameter of the sun itself, and the near edge of the triangle will be formed by the image of the sun on another index card.



Hold the two cards parallel, with the pinhole card closest the sun (see the above picture). You will need to record two numbers: the size of the sun's image on the lower card, and the distance between the two cards.

This is the same way that you make a pinhole camera (packing



it all in a nice dark box). You will need a sunny day to do this. If it's cloudy, see the next chapter to do the adventure on identifying cloud types until it clears up.

The program for this adventure needs the distances you've measured, and it will then tell you the distance from you to the sun. For some "further adventures," try this several times, and take an average (to reduce the effect of any errors in your measurements). Have your friends do the same, and try to get as close as possible to a textbook value for the distance. How do you get a textbook value? Why, you go to your nearest encyclopedia, of course!

Program Listing

```
10 HOME
20 PRINT "      DISTANCE TO THE SUN"
30 PRINT
40 PRINT
50 PRINT "USING THE CARD ACTIVITY"
60 PRINT "MENTIONED IN THE TEXT YOU"
70 PRINT "CAN ESTIMATE THE DISTANCE FROM"
80 PRINT "YOUR SPOT ON EARTH TO THE SUN."
```



```
90 PRINT
100 PRINT "WHAT IS THE DIAMETER OF THE"
110 PRINT "SUN'S IMAGE ON THE CARD"
120 PRINT "(IN CM)?"
130 INPUT d
140 PRINT
150 PRINT "WHAT IS THE DISTANCE FROM THE "
160 PRINT "PINHOLE CARD TO THE IMAGE"
170 PRINT "CARD (IN CM)?"
180 INPUT s
190 ds = s*1392000/d
200 p = ABS((150000000-s*1392000/d)/150000000)*100
210 ds = INT(ds*10^3+.5)/INT(10^3+.5)
220 p = INT(p*10^2+.5)/INT(10^2+.5)
230 PRINT
240 PRINT "THE ESTIMATED DISTANCE TO THE"
250 PRINT "SUN IS "; ds; " KM."
260 PRINT "YOUR PERCENTAGE OF ERROR IS "
270 PRINT p; "%."
280 IF p > 20 THEN 350
290 PRINT
300 PRINT "THAT'S A PRETTY GOOD RESULT."
310 PRINT "CHECK THAT AGAINST THE AVERAGE"
320 PRINT "DISTANCE WHICH IS 150,000,000"
330 PRINT "OR 1.5 X 10^8 KILOMETERS."
340 GOTO 380
350 PRINT
360 PRINT "WHY DON'T YOU TAKE THOSE"
370 PRINT "MEASUREMENTS OVER AGAIN."
380 END
```

Explanation

The variable list is shown here. The line-by-line explanation follows.

d	diameter of the sun image on the card
s	distance from pinhole to image
ds	calculated sun distance
p	percent error of ds as compared to the actual value

Lines 10-90. Prints the introduction.

Lines 100-130. Asks for the diameter of the sun's image and the pinhole card.

Lines 140-180. Asks for the distance between the image and the pinhole card.

Line 190. Calculates distance to the sun based on the ratio of the image to the distance (sun dist/sun diam = 107.758) and the actual diameter (1392000 km).

Line 200. Calculates the percent error by comparing your results with the known results (textbook values).

Lines 210-220. Rounds distance to three decimal places and error percentage to two decimal places.

Lines 230-270. Displays the results.

Line 280. Checks to see if your percentage of error is high (20 percent or more). If it is, the program goes to line 350. If not, it goes to 290.

Lines 290-340. These print a message if your percentage of error is less than 20 percent, and then end the program.

Lines 350-370. If error is greater than 20 percent, suggests that you try again.

Line 380. The end.

Sample Run

```
      DISTANCE TO THE SUN


USING THE CARD ACTIVITY
MENTIONED IN THE TEXT YOU
CAN ESTIMATE THE DISTANCE FROM
YOUR SPOT ON EARTH TO THE SUN.

WHAT IS THE DIAMETER OF THE
SUN'S IMAGE ON THE CARD
(IN CM)?
?.5

WHAT IS THE DISTANCE FROM THE
PINHOLE CARD TO THE IMAGE
CARD (IN CM)?
?55

THE ESTIMATED DISTANCE TO THE
SUN IS 153120000 KM.
YOUR PERCENTAGE OF ERROR IS
2.08%.

THAT'S A PRETTY GOOD RESULT.
CHECK THAT AGAINST THE AVERAGE
DISTANCE WHICH IS 150,000,000
OR 1.5 X 10^8 KILOMETERS.
```


3

Adventures in Earth Science

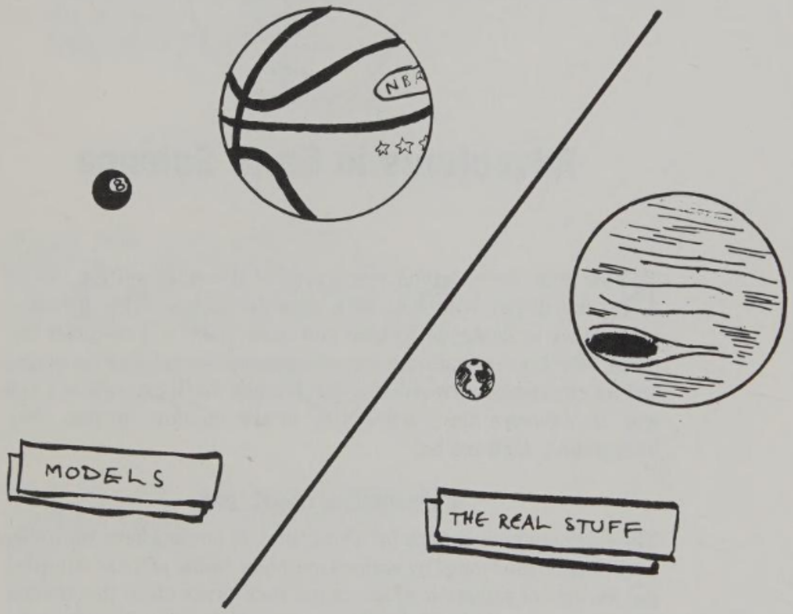
Now that we've given you a tour of the solar system, we're going to get your feet back on solid ground. This group of adventures is designed to help you understand and decipher the earth. We'll investigate the way it operates, its makeup (geology) and its environment (weather and climates). We'll even show you a way to measure the earth's size in the comfort of your own background! Onward ho!

A GEOLOGICAL TIME CLOCK

The earth has been around for a long time. Scientists have been able to estimate "how long" by various methods. Some of these compare the amount of radiation which an old rock gives off to the amount which a new rock gives off. This gives us a clue as to how long the rock has been radiating (its age). The oldest rocks found on the earth are roughly four billion years old. A billion is a lot. For instance, it's been about a billion seconds since World War II. It's

been a billion minutes since we began counting Years A.D. That's getting a little hard to picture.

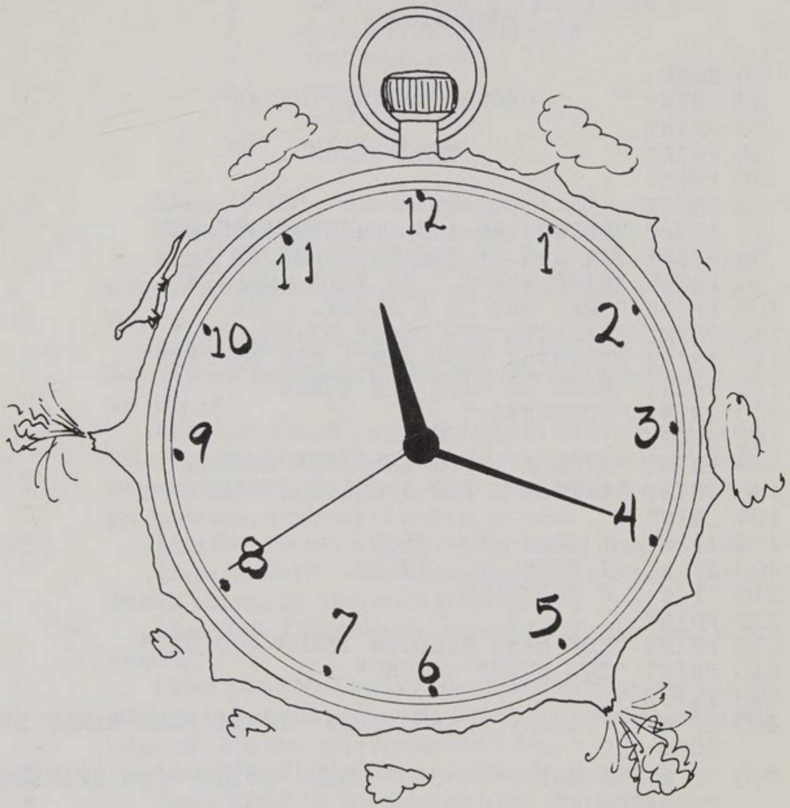
Let's use a trick we used to understand distances in the first chapter. We'll make a model. So far, the models we have used have helped us to scale distances. We have simply used things (golf balls,



frisbees, etc.) which are smaller than, but similar to the real objects (Mars, Saturn, etc.).

This model will be built the same way. Since we're trying to understand them, let's get some smaller time. Since you all have a fairly good idea of how long a day is, let's scale our earth's history so that the entire four billion years fits into a single 24-hour day.

We have already taught ADAM how to do this scaling. That's what the program does. What you need to tell it is how long ago, in millions of years, an event occurred. ADAM will then tell you, that if the entire history of the earth equals 24 hours, your event occurred at a certain clock time (AM or PM).



Here are a few times to try. For others, you can find the number of years ago they happened in an encyclopedia.

Event	How Long Ago (approximately)
First rocks cooled	3800 million years
Water formed	3000
First life forms	570
First insects	375
First fishes	400
Ice Age	2
Appalachians formed	225
Rocky mountains formed	70
Humans appeared on earth	.05

Program Listing

```
 10 HOME
 20 PRINT "     GEOLOGICAL/UNIVERSAL"
 30 PRINT
 40 PRINT "          TIME CLOCK"
 50 PRINT
 60 PRINT
 70 PRINT "SELECT AN AGE OF THE EARTH OR"
 80 PRINT "AN AGE OF THE UNIVERSE AS A "
 90 PRINT "TIME SCALE.  WE WILL COMPARE"
100 PRINT "THAT AGE TO A CLOCK.  THE "
110 PRINT "EARLIER THE CLOCK READS, FOR"
120 PRINT "EXAMPLE 1:00 A.M., THE FURTHER"
130 PRINT "BACK IN TIME THE EVENT"
140 PRINT "OCCURRED."
150 PRINT
160 PRINT "TYPE A 1 FOR AN EARTH AGE OR"
170 PRINT "TYPE A 2 FOR A UNIVERSE AGE."
180 INPUT a
190 IF a = 1 THEN  b = 4600
200 IF a = 2 THEN  b = 18000
210 IF a > 2 THEN 160
220 PRINT
230 PRINT "HOW MANY MILLION YEARS AGO DID"
240 PRINT "THE EVENT OCCUR?"
250 INPUT y
260 IF a = 1 AND y > 4600 THEN  PRINT "THE EARTH DID NOT EXIST!":
    GOTO 150
270 IF a = 2 AND y > 18000 THEN  PRINT "THE UNIVERSE DID NOT EXIST!
    ": GOTO 150
280 r = 24-24*y/b
290 h = INT(r)
300 q = 60*(r-h)
310 m = INT(q)
320 s = INT(60*(q-m))
330 m$ = "A.M."
340 IF h = 12 THEN  m$ = "P.M."
350 IF h > 12 THEN  m$ = "P.M.": h = h-12
360 IF h = 0 THEN  h = 12
370 PRINT "THE EVENT OCCURRED AT ABOUT "
380 PRINT h; ":"; m; ":"; s; " "; m$
390 END
```

Explanation

The variables and the line-by-line explanations are given here.

a	universal or geological
b	full age of earth or universe (in millions of years)
y	time (in millions of years) of event
r	decimal clock time of event
h	clock hours in r
q	minutes in r
s	seconds in r
m$	AM/PM indicator

Lines 10-150. Prints the introduction and instructions.

Lines 160-180. Asks use of earth or universe clock.

Lines 190-210. Sets full-scale age of earth (b=4600 million years) or universe (18,000 million years). Asks again if a is not 1 or 2.

Lines 220-270. Accepts a value for the event time (in millions of years) and checks to make sure that your event time is not too long ago.

Line 280. Calculates y/b, the ratio of the event time to the full-scale age of the earth or universe, multiplies it by 24 to get hours-ago on the 24-hour scale, and then subtracts that from 24 to get hours-since-beginning of earth or universe.

Line 290. Takes the integer of that number to get hours (h).

Lines 300-310. Scales to noninteger part of decimal time to decimal minutes (q), then takes its integer (m).

Line 320. Calculates the second left(s) after calculating minutes.

Lines 330-350. Checks to see if the time is PM (hours 12 or greater), and sets m$ to AM or PM. If the hours are greater than 12, subtracts 12 to stay away from military time.

Line 360. If in the hour past midnight, changes to 0 to 12.

Lines 370-380. Prints out the clock time of the event.

Sample Run

```
     GEOLOGICAL/UNIVERSAL

          TIME CLOCK

SELECT AN AGE OF THE EARTH OR
AN AGE OF THE UNIVERSE AS A
TIME SCALE.  WE WILL COMPARE
THAT AGE TO A CLOCK.  THE
EARLIER THE CLOCK READS, FOR
EXAMPLE 1:00 A.M., THE FURTHER
BACK IN TIME THE EVENT
OCCURRED.

TYPE A 1 FOR AN EARTH AGE OR
TYPE A 2 FOR A UNIVERSE AGE.
?1
```

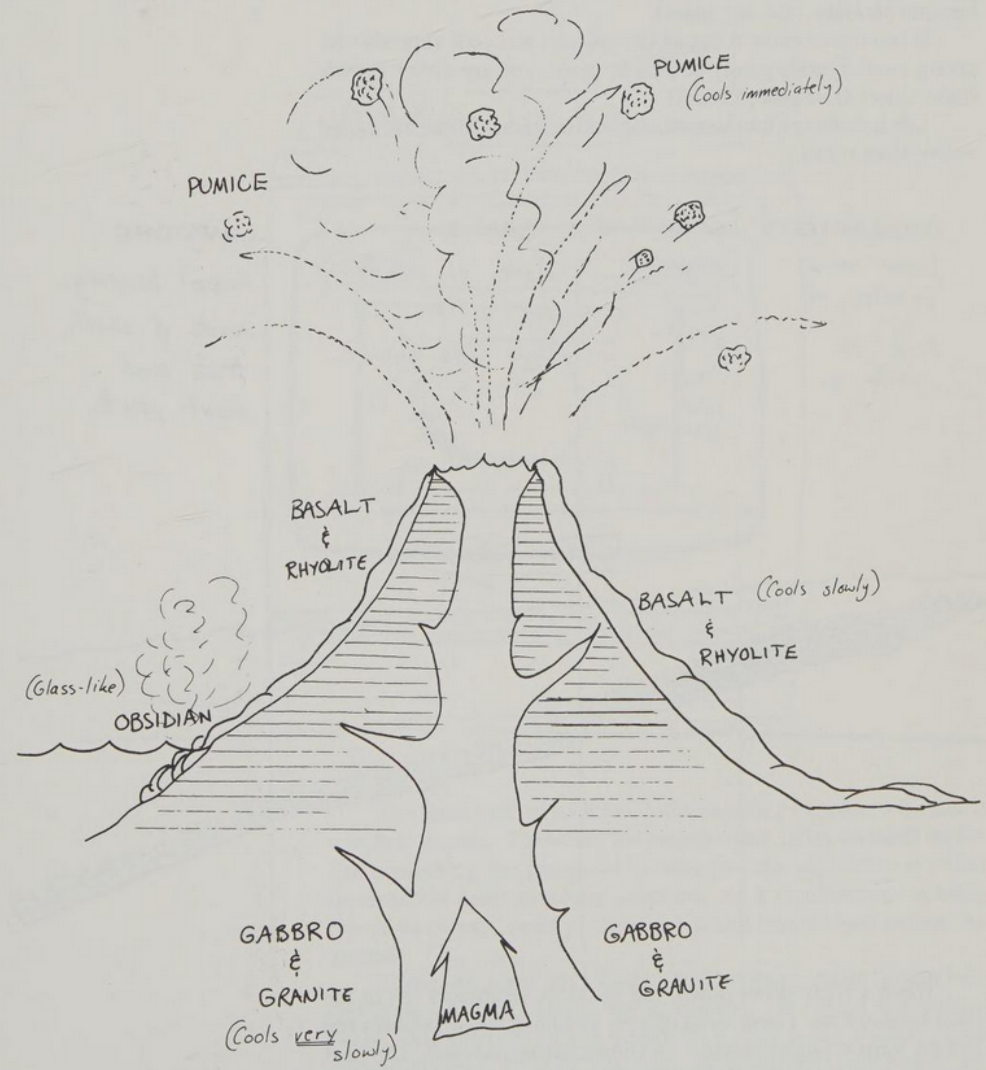
```
HOW MANY MILLION YEARS AGO DID
THE EVENT OCCUR?
?240
THE EVENT OCCURRED AT ABOUT
10:44:52 P.M.
```

ROCK IDENTIFICATION

Wouldn't you like to know the life story of the next rock you stub your toe on? It may not make you feel any better, but you'll be able

to appreciate what the rock's been through. Some of these rocks have had a pretty rough history. Consider the possibilities: a rock can form from inside of a volcano, from being dumped at the bottom of a river, or from being crushed under miles of other rock or ice. None of these sound too appealing.

But that's the way a rock's existence is. To get technical, the rocks which formed in volcanoes are known as *igneous* (fire-formed) rocks. If you start in a volcano, there are four possibilities for where you end up.



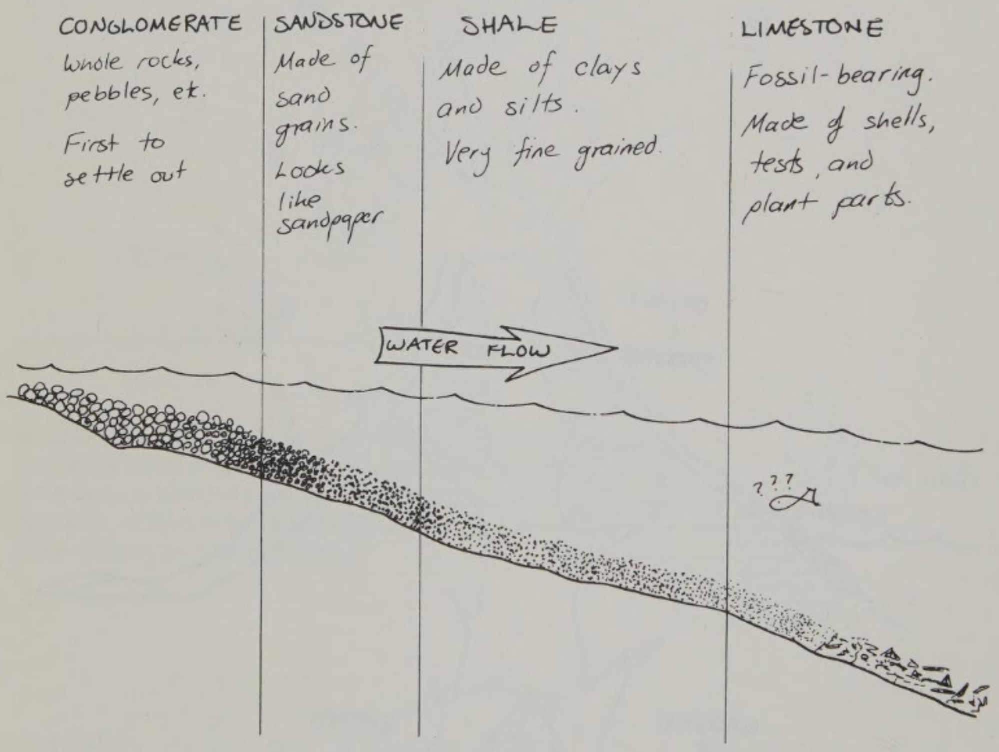
If you start as molten rock (magma) and are thrown into the air and cool so quickly that your bubbles remain intact, you're known as *pumice*.

If you start as magma and cool slowly on the slopes of the volcano, you are *basalt* (if your crystals are dark colors), or *rhyolite* (if your crystals are light).

If you start as magma and make it down the slope of the volcano, only to run into some water, you cool so quickly that you become *obsidian* (volcano glass).

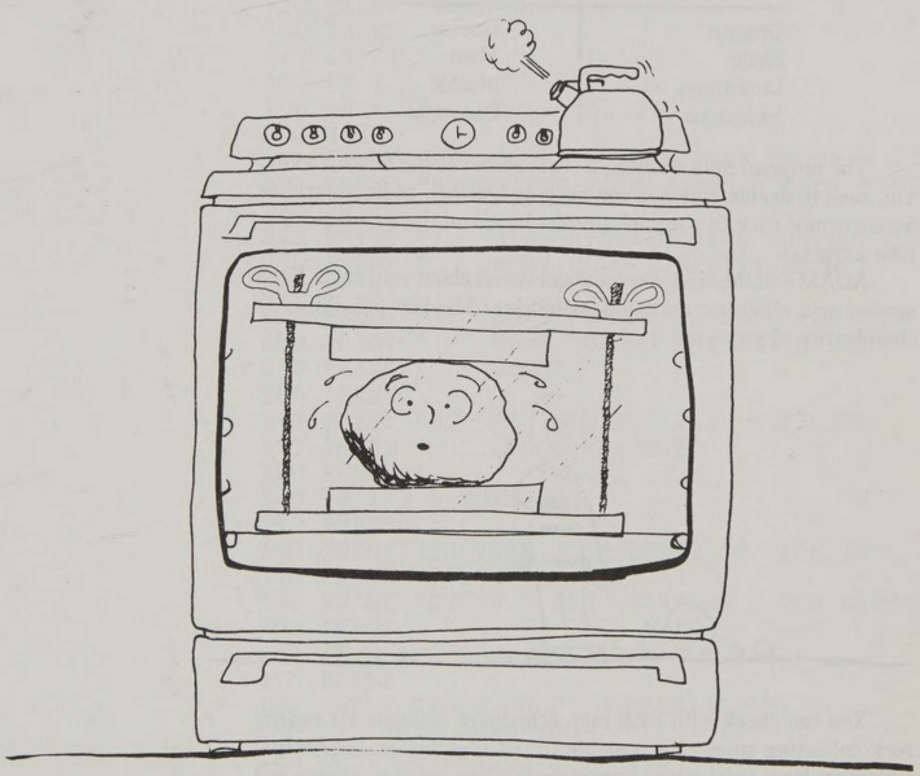
If you never make it out of the volcano and cool very slowly, giving your crystals plenty of time to grow, you are either *granite* (light colored) or *gabbro* (dark).

Life is different for the rocks formed in water. These are called *sedimentary* rocks.



When a river slows down, the heaviest sediments fall to the river bottom first. These large pieces, including whole pebbles and stones, form a rough-grained rock known as *conglomerate*. The next particles to fall out are the sand-sized particles, which end up as *sandstone*. The finest-grained sediments are what we call clays and silts (you probably call them mud). The rock which forms from these smallest particles is called *shale*.

There's one more variety of sedimentary rock: those which form from the remains of dead animals, seashells, or plant parts living in an ocean or river are called *limestone*. If some of the shells come through without being crushed, we call them *fossils*.



The third type of rock has had something happen to it after it was first formed. These are the *metamorphic* (after-formed) rocks. The something that happened to them was the application of either tremendous heat, crushing pressure, or a combination of both. Needless to say, most of these are found buried deep within the ground.

There are some very famous metamorphic rocks. If you heat and crush dead dinosaurs, trees, mosses, etc.. for a few million years, you get coal (if it's originally mostly trees, without much decomposition). If the original material contained more water, and underwent bacterial decomposition, you'll get oil. (In a manner of speaking, the energy crisis exists because we're running out of dead dinosaurs . . .)

Since most metamorphic rocks used to be some other rock, we can show you the relationships between the originals and the end-products.

Used To Be	Became
Granite	Gneiss
Shale	Slate
Limestone	Marble
Sandstone	Quartzite

The program puts all these facts together in the form of a key. You need to decide whether you have an igneous, sedimentary, or metamorphic rock (if you can't predict based on the above stories, take a guess).

ADAM will then ask you some questions about your rock. Each answer will eliminate certain possibilities. The last possibility is the identity of your rock.

You can check with local high schools or colleges for nearby rock collecting areas. It's possible to put together an impressive collection from local sites. Newspapers often list rock shows and swap meets, which allow you to get rocks which are not available locally.

Happy hunting!

Program Listing

```
10 HOME
20 PRINT "     IDENTIFY THAT ROCK!"
30 PRINT
40 PRINT
50 PRINT
60 PRINT "WHAT TYPE OF ROCK DO YOU HAVE?"
70 PRINT
80 PRINT "1.  IGNEOUS"
90 PRINT
```



```
100 PRINT "2.  SEDIMENTARY"
110 PRINT
120 PRINT "3.  METAMORPHIC"
130 PRINT
140 PRINT "TYPE IN THE NUMBER OF YOUR "
150 PRINT "CHOICE."
160 PRINT
170 INPUT t
180 IF t > 3 THEN 130
190 ON t GOTO 200, 520, 850
200 HOME
210 PRINT "ENTER THE NUMBER THAT BEST"
220 PRINT "DESCRIBES YOUR IGNEOUS ROCK."
230 PRINT
240 PRINT "1.  BIG CRYSTALS WITH TWO OR"
250 PRINT "    MORE COLORS EASILY SEEN"
260 PRINT
270 PRINT "2.  SMALL CRYSTALS WITH TWO OR"
280 PRINT "    MORE COLORS BUT NOT EASILY"
290 PRINT "    SEEN"
300 PRINT
310 PRINT "3.  NO CRYSTALS AND ONLY ONE"
320 PRINT "    COLOR PRESENT"
330 INPUT i
340 IF i > 3 THEN 200
350 PRINT
360 PRINT "IS YOUR ROCK LIGHT OR DARK IN"
370 PRINT "COLOR?"
380 PRINT "ENTER 1 FOR LIGHT OR 2 FOR DARK"
390 INPUT c
400 IF c > 2 THEN 360
410 PRINT
420 PRINT "YOUR ROCK IS PROBABLY ";
430 IF c = 2 THEN 480
440 ON i GOTO 450, 460, 470
450 PRINT "GRANITE.": END
460 PRINT "RHYOLITE.": END
470 PRINT "PUMICE.": END
480 ON i GOTO 490, 500, 510
490 PRINT "GABBRO.": END
500 PRINT "BASALT.": END
510 PRINT "OBSIDIAN.": END
520 HOME
530 PRINT "ENTER THE NUMBER WHICH BEST"
540 PRINT "DESCRIBES YOUR SEDIMENTARY"
550 PRINT "ROCK'S TEXTURE."
560 PRINT
570 PRINT "1.  FEELS COARSE OR ROUGH."
580 PRINT "    PEBBLES AND ROCK FRAGMENTS"
590 PRINT "    ARE VISIBLE."
```

```
600 PRINT
610 PRINT "2.  FEELS SOMEWHAT ROUGH."
620 PRINT "    SAND-SIZE PARTICLES ARE"
630 PRINT "    PRESENT."
640 PRINT
650 PRINT "3.  FEELS SMOOTH AND FINE."
660 PRINT "    LOOKS LIKE THE ROCK IS"
670 PRINT "    MADE OF THIN LAYERS OF"
680 PRINT "    MUD."
690 PRINT
700 PRINT "4.  PARTS OF SEA ANIMALS OR"
710 PRINT "    PLANTS ARE VISIBLE."
720 PRINT "    SEASHELLS MAY BE PRESENT."
730 PRINT
740 PRINT "ENTER THE NUMBER OF YOUR"
750 PRINT "CHOICE."
760 INPUT s
770 IF s > 4 THEN 520
780 PRINT "YOUR ROCK IS PROBABLY ";
790 ON s GOTO 800, 820, 830, 840
800 PRINT
810 PRINT "CONGLOMERATE.": END
820 PRINT "SANDSTONE.": END
830 PRINT "SHALE.": END
840 PRINT "LIMESTONE.": END
850 HOME
860 PRINT "ENTER THE NUMBER WHICH BEST"
870 PRINT "DESCRIBES YOUR METAMORPHIC"
880 PRINT "ROCK."
890 PRINT
900 PRINT "1.  MICA LAYERS ARE PRESENT IN"
910 PRINT "    THE ROCK."
920 PRINT
930 PRINT "2.  THE ROCK LOOKS LIKE IT IS"
940 PRINT "    MADE UP OF LIGHT AND "
950 PRINT "    DARK STRIPES."
960 PRINT
970 PRINT "3.  THE ROCK IS LAYERED AND"
980 PRINT "    VERY SMOOTH.  IT LOOKS"
990 PRINT "    LIKE IT IS MADE OF"
1000 PRINT "    LAYERS OF MUD."
1010 PRINT
1020 PRINT "4.  THE ROCK IS SMOOTH, NOT"
1030 PRINT "    GRAINY.  IT LOOKS LIKE"
1040 PRINT "    QUARTZ."
1050 PRINT
1060 PRINT "ENTER THE NUMBER OF YOUR"
1070 PRINT "CHOICE."
1080 PRINT
1090 INPUT m
1100 IF m > 4 THEN 850
```



```
1110 PRINT
1120 PRINT "YOUR ROCK IS PROBABLY ";
1130 ON m GOTO 1140, 1150, 1170, 1190
1140 PRINT "SCHIST.": END
1150 PRINT "GNEISS."
1160 PRINT "GNEISS WAS ONCE GRANITE.": END
1170 PRINT "SLATE."
1180 PRINT "SLATE WAS ONCE SHALE.": END
1190 PRINT "QUARTZITE.": END
```

Explanation

The variable definitions are listed here. The line-by-line definitions follow.

t	number of major group
i	types of crystals (igneous)
c	light or dark color (igneous)
s	appearance of sedimentary rock
m	appearance of metamorphic rock

Lines 10-190. These lines make up the first menu of choices. Line 190 sends control to the other three major parts of the program depending on your value for t.

Lines 200-340. These lines set a value, i, which isn't used for a while. It will be used only when the color of the crystals is known, which is done in 350-430.

Lines 350-430. These lines accept another value, c, which represents the color of your rock.

Lines 440-470. The program is sent here if the crystals are light (c=1), and goes to 450, 460, or 470 depending on crystal type (1, 2, or 3).

Lines 480-510. This does the same as 440-470, but for dark (c=2) crystals.

Lines 520-780. Menu of choices for appearance of sedimentary rocks.

Lines 790-840. Sends control to correct sedimentary name.

Lines 850-1090. Menu of choices for metamorphic rock.

Lines 1100-1200. Sends control to correct metamorphic name.

Sample Run

```
                IDENTIFY THAT ROCK!


WHAT TYPE OF ROCK DO YOU HAVE?
```

```
1.   IGNEOUS

2.   SEDIMENTARY

3.   METAMORPHIC

TYPE IN THE NUMBER OF YOUR
CHOICE.

?1
ENTER THE NUMBER THAT BEST
DESCRIBES YOUR IGNEOUS ROCK.

1.   BIG CRYSTALS WITH TWO OR
     MORE COLORS EASILY SEEN

2.   SMALL CRYSTALS WITH TWO OR
     MORE COLORS BUT NOT EASILY
     SEEN

3.   NO CRYSTALS AND ONLY ONE
     COLOR PRESENT
?2

IS YOUR ROCK LIGHT OR DARK IN
COLOR?
ENTER 1 FOR LIGHT OR 2 FOR DARK
?1

YOUR ROCK IS PROBABLY RHYOLITE.
```

CLOUD IDENTIFICATION

When you lie under the open sky and look up at the clouds, you usually find yourself seeing things in the clouds. One cloud may look like a duck, another like a dragon, and yet another like Uncle Ralph's profile.

The clouds seem to be rather whimsical, changing and moving in whatever ways the winds pull them. After a while, it begins to look like a pretty disorganized show.

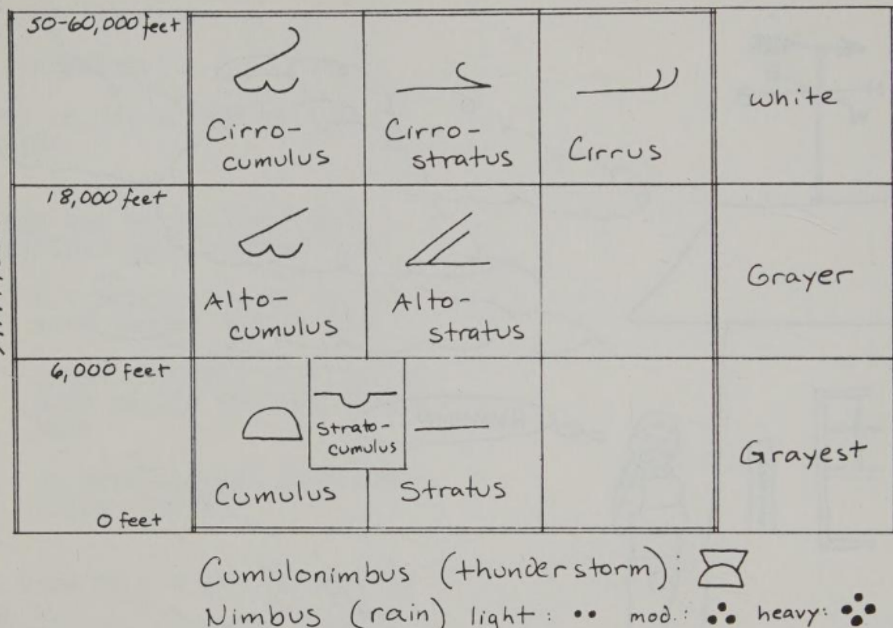
On the contrary, clouds can tell you a great deal about what's going on up there. Meteorologists use the number and type of clouds to help them understand and predict weather patterns.

If you're judging clouds by what you've seen lying out in a field, that's not fair. You've never seen rainclouds that way (we hope), and you've certainly never seen thunderclouds from that viewpoint. You may have ignored some of the more important clouds which, to the daydreaming eye, look pretty boring.

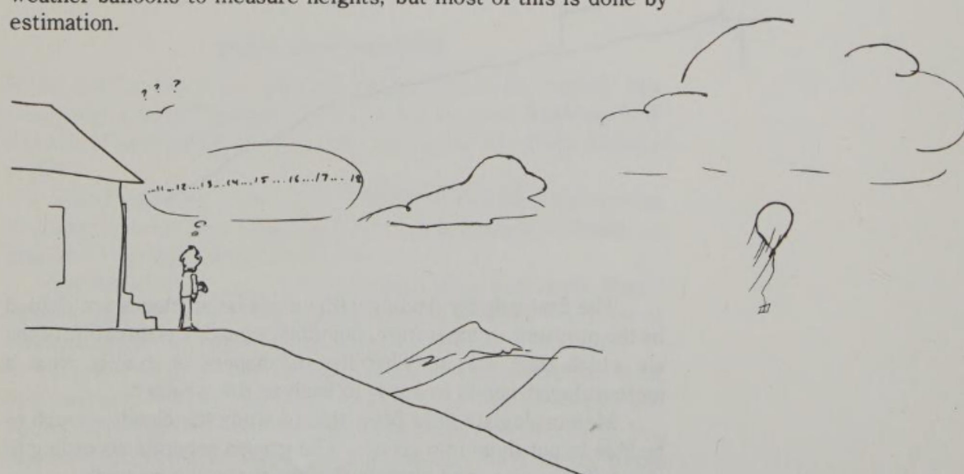
The first rule for dealing with clouds is: all clouds are shaped by the pressure, temperature, humidity, speed, and direction of the air which they are in. That list of shapers is exactly what a meteorologist needs to know to analyze the weather.

Meteorologists have been able to study the clouds enough to be able to put them into groups. The groups separate according to cloud shape, size, and altitude (height above the ground).

Here's what the groups look like if you arrange them in a table:



As you can see, the cloud's identity depends on what it looks like and where it is. Those two things are fairly easy to find out with the naked eye. Some meteorologists who need exact altitudes use weather balloons to measure heights, but most of this is done by estimation.

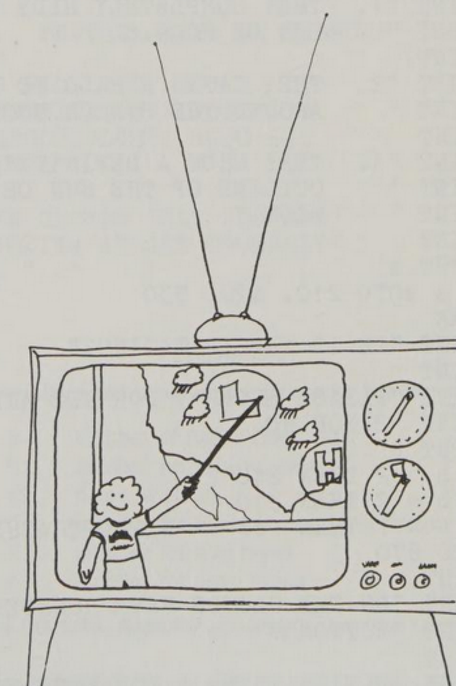


The program in this adventure will do the identifying for you. This program acts as a key, similar to the one in the last adventure on identifying rocks. When you RUN this program, ADAM will ask you some specific questions about the cloud(s) that interest you. A nice setup would be to have ADAM near a window, making it easier to check on the clouds. If that's not possible, ADAM's infinite patience will allow you to run to the window or outside to check.

Once you run this program a few times, you'll begin to see that it is organized just like our table. The clouds' characteristics dictate what kind it is. All clouds with rounded tops are ***cumulo***–something-or-other; sheet-like clouds have ***stratus*** in their names; and anything that has something falling from it (rain, snow, sleet, hail, etc.) gets ***nimbus*** tagged on it.

With a little practice, you'll be able to identify clouds by sight. Use this key as a refresher or when stumped by a particularly odd-looking cloud.

If you'd like to take things one step further and see what all of these clouds look like from the top down, watch your tv stations weather report. Many stations now carry satellite photos of the



day's weather. You can check for the cloud cover in your area, and compare it to your observations.

In our next adventure, we will need to know the cloud types in order to predict the weather for the next 12 hours. So polish up your cloud identification skills and match wits with the pros!

Please be careful when doing any activities concerning severe weather. While thunderheads are fascinating to look at, that's best done from indoors. Always keep informed of the forecast during times of dangerous weather by listening to national or local weather service reports. When they tell you to watch out, listen!

Program Listing

```
 10 HOME
 20 PRINT "     IDENTIFY THE CLOUDS!"
 30 PRINT
 40 PRINT
 50 PRINT "ENTER THE NUMBER THAT BEST"
 60 PRINT "DESCRIBES THE CLOUDS YOU ARE"
 70 PRINT "IDENTIFYING."
 80 PRINT
 90 PRINT "1.  THEY COMPLETELY HIDE THE"
100 PRINT "    SUN OR MOON."
110 PRINT
120 PRINT "2.  THEY CAUSE A HALO TO FORM"
130 PRINT "    AROUND THE SUN OR MOON."
140 PRINT
150 PRINT "3.  THEY SHOW A DEFINITE"
160 PRINT "    OUTLINE OF THE SUN OR"
170 PRINT "    MOON."
180 PRINT
190 INPUT a
200 ON a GOTO 210, 420, 530
210 HOME
220 PRINT "IS IT PRECIPITATING?"
230 PRINT
240 PRINT "PLEASE ENTER 1 FOR YES AND"
250 PRINT "2 FOR NO."
260 INPUT b
270 IF b > 2 THEN 240
280 IF b = 2 THEN 310
290 IF b = 1 THEN  c$ = "NIMBOSTRATUS OR CUMULONIMBUS"
300 GOTO 670
310 PRINT
320 PRINT "DO THE CLOUDS HAVE ROUNDED"
330 PRINT "BOTTOMS?"
340 PRINT
350 PRINT "PLEASE ENTER 1 FOR YES AND"
360 PRINT "2 FOR NO."
```



```
370 INPUT c
380 IF c > 2 THEN 350
390 IF c = 1 THEN  c$ = "CUMULUS OR STRATOCUMULUS"
400 IF c = 2 THEN  c$ = "STRATUS"
410 GOTO 670
420 HOME
430 PRINT "ARE THE CLOUDS SHEETLIKE OR IN"
440 PRINT "LAYERS?"
450 PRINT
460 PRINT "PLEASE ENTER 1 FOR YES AND"
470 PRINT "2 FOR NO."
480 INPUT d
490 IF d > 2 THEN 460
500 IF d = 1 THEN  c$ = "ALTOSTRATUS"
510 IF d = 2 THEN  c$ = "ALTOCUMULUS"
520 GOTO 670
530 HOME
540 PRINT "ENTER THE NUMBER THAT DESCRIBES "
550 PRINT "THE CLOUDS."
560 PRINT
570 PRINT "1.  LAYERED"
580 PRINT
590 PRINT "2.  BUMPY BOTTOMED"
600 PRINT
610 PRINT "3.  FEATHER-LIKE OR FIBROUS"
620 INPUT e
630 ON e GOTO 640, 650, 660
640 c$ = "CIRROSTRATUS": GOTO 670
650 c$ = "CIRROCUMULUS": GOTO 670
660 c$ = "CIRRUS"
670 HOME
680 PRINT "THE CLOUDS THAT YOU ARE"
690 PRINT "LOOKING AT ARE PROBABLY"
700 PRINT c$; " ."
710 END
```

Explanation

The variables and program explanations follow:

a	number of major cloud type
b	number for rain/no rain
c$	name of cloud
c	number for cloud bottom description
d	number for alto types
e	number for cirro types

Lines 10-180. Asks for general description of clouds in the sky.

Lines 190-200. Sends control to the appropriate routine based on value of a.

Lines 210-290. Identifies nimbus-type clouds based on rain or no rain. b holds the value for rain/no rain.

Lines 300-410. Identifies thick clouds as cumulus or stratus based on c, the shape of the cloud bottoms.

Lines 420-520. For high clouds, selects cumulus or stratus based on sheets (d=1) or no sheets (d=2).

Lines 530-700. For well-defined, medium altitude clouds, this routine selects cirrus-type clouds on the basis of layering (e=1), rounded shape (e=2), or filament form (e=3).

Sample Run

```
      IDENTIFY THE CLOUDS!


ENTER THE NUMBER THAT BEST
DESCRIBES THE CLOUDS YOU ARE
IDENTIFYING.

1.   THEY COMPLETELY HIDE THE
     SUN OR MOON.

2.   THEY CAUSE A HALO TO FORM
     AROUND THE SUN OR MOON.

3.   THEY SHOW A DEFINITE
     OUTLINE OF THE SUN OR
     MOON.

?1
IS IT PRECIPITATING?

PLEASE ENTER 1 FOR YES AND
2 FOR NO.
?1
THE CLOUDS THAT YOU ARE
LOOKING AT ARE PROBABLY
NIMBOSTRATUS OR CUMULONIMBUS .
```

TWELVE-HOUR FORECAST

Now that we know "what's up" concerning clouds, (you did read the last adventure, didn't you?), we can use our new-found expertise to analyze and predict the local weather.

Just as clouds seem erratic and confusing at first, weather patterns seem to vary wildly. If you live in certain parts of the

country, such as New England or the Midwest, this can be doubly so.

Actually, most of the time a forecaster's life is much simpler than it seems. Just as clouds have certain things in common that allow them to be grouped, weather patterns can be narrowed down to a few common types. Even the ones which seem to be rather disturbing. Hurricanes, tropical storms, and tornadoes can be easily

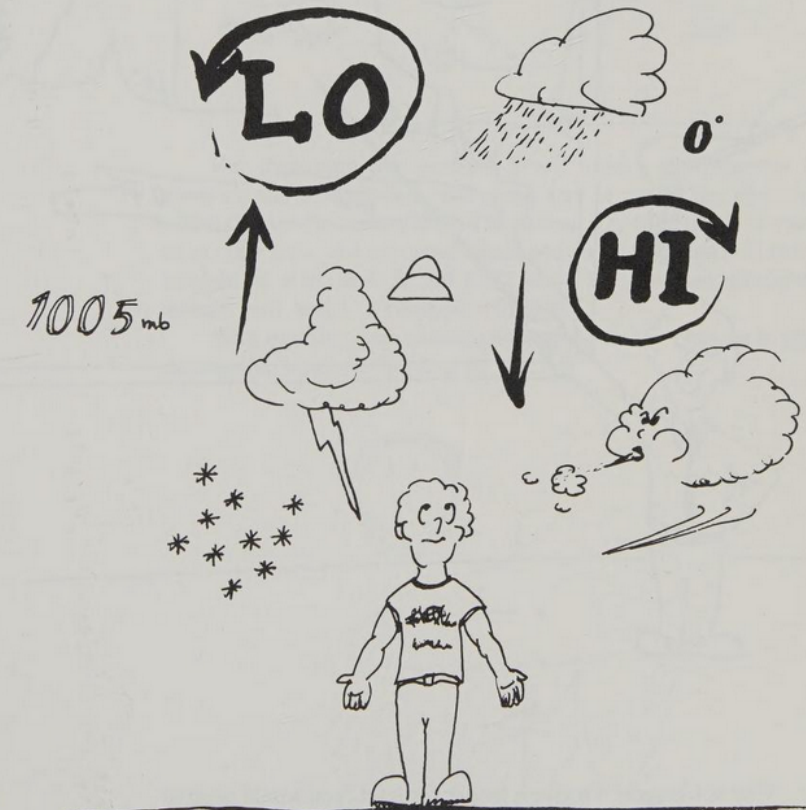
identified and predicted. Most of the time. For the sake of simplicity, we'll leave those out of this discussion,

The greatest controller of weather patterns is the air's pressure. It's the relative changes in pressure that create weather, wind, and precipitation patterns. Pressure in a certain region can either be high or low (compared to what else is around it). So we speak of a *region of low pressure* (from this point on, we'll simply call one of these a LO) or a *region of high pressure* (simply called a HI).

When air begins to move up due to its low pressure or down due to its high pressure, the spin of the earth causes it to *spiral* up or down rather than move straight up or down. (This is known as the *Coriolis Effect*, and it's the same thing that gives a spin to water going down the drain.)

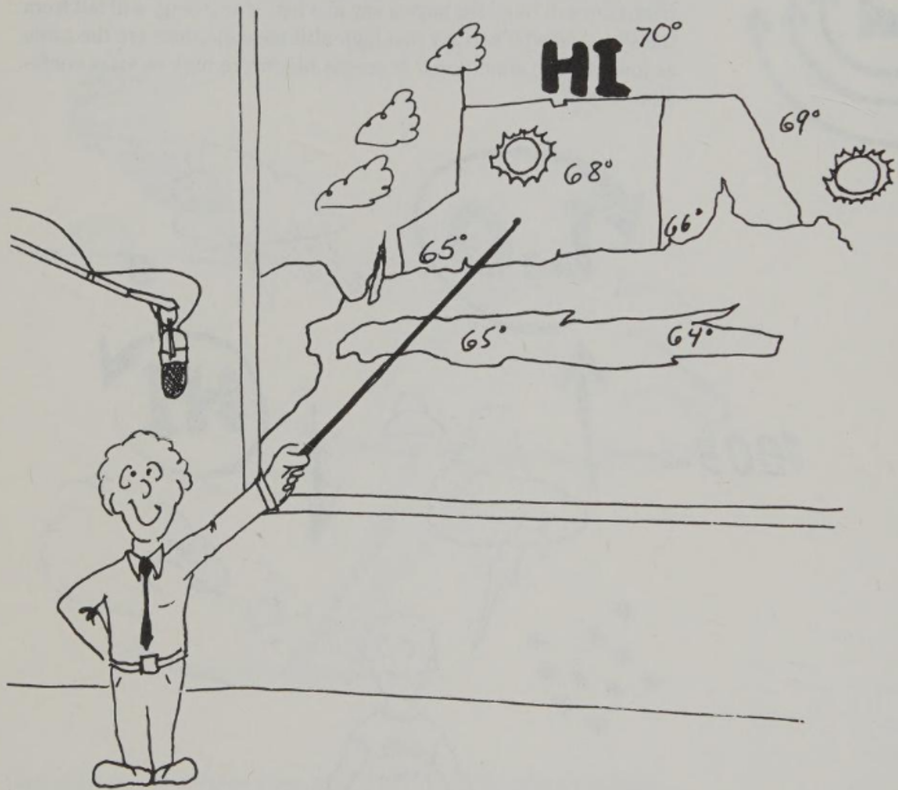
This means that we can say that in every HI, the winds end up moving *clockwise*. Conversely, the winds in a LO rotate *counterclockwise*. Simple, no?

Well, not that simple. If the *temperature* changes, it will heat or cool the air and make it less or more likely to precipitate. Temperature will also dictate what type of precipitation you will have. *Humidity* will likewise have a say in what, if anything, will fall from the sky. And who's to say that high-altitude conditions are the same as low-altitude conditions? It seems like we're back to mass confusion, right?

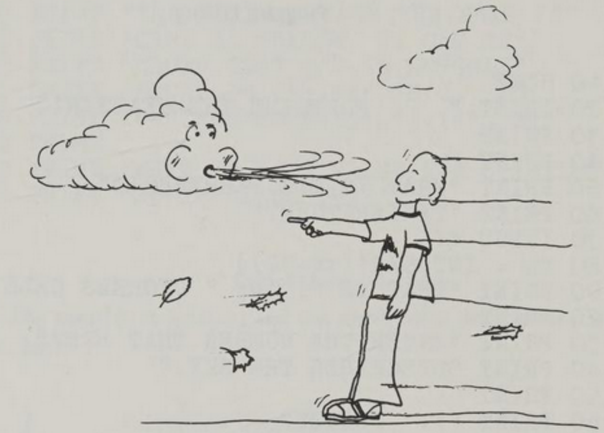


Not really. We already have a clue to all of those conditions which we just mentioned. As we saw in our last adventure, the clouds are shaped by temperature, humidity, altitude, and pressure. Therefore, if you know the clouds, you have an excellent way of knowing the conditions most of the time.

All of this knowledge enables us to predict the weather for the next 12 hours based on the temperature, wind direction, and current conditions. The program is written so that ADAM will ask you for all of these things, and give you the forecast for the next 12 hours.

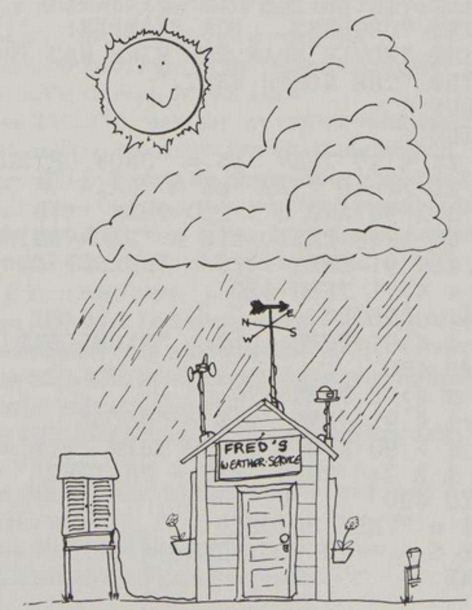


Why not longer? To give a longer forecast, you would need to know what conditions are like wherever the winds and weather are coming from. That's precisely what the National Weather Service is: a network of observers who share enough information to provide a complete picture of the weather across the country and beyond. While it's hard for you to get on this network, there's no reason why you and a handful of friends around your state or region can't set up your own network to provide your school or club with forecasts.



For observing the weather, you'll need a thermometer to measure the temperature, and some way of noting the direction which the wind is coming from. The easiest way to do this is to stand in an open area, and turn your whole body until the wind is blowing straight at your face. If you don't know the compass directions around your house, a compass will help.

With practice, you should get as good as the experts in predicting the weather. Good luck!



Program Listing

```
10 HOME
20 PRINT "     FORECAST THE WEATHER!"
30 PRINT
40 PRINT
50 PRINT "WHAT IS THE FAHRENHEIT"
60 PRINT "TEMPERATURE?"
70 INPUT tf
80 tc = INT(5/9*(tf-32))
100 PRINT "THAT IS "; tc; " DEGREES CELSIUS."
120 PRINT
130 PRINT "ENTER THE NUMBER THAT BEST"
140 PRINT "DESCRIBES THE SKY."
150 PRINT
160 PRINT "1.  CLOUDY"
170 PRINT "2.  PARTLY CLOUDY"
180 PRINT "3.  CLEAR"
190 INPUT s
200 IF s > 3 THEN 130
210 PRINT
220 PRINT "FROM WHAT DIRECTION IS THE"
230 PRINT "WIND COMING? "
240 PRINT
250 PRINT "E=90   S=180   W=270   N=360"
260 PRINT
270 PRINT "ENTER THE DIRECTION IN"
280 PRINT "DEGREES.  FOR EXAMPLE:  315"
290 PRINT "WOULD MEAN THE WIND WAS FROM"
300 PRINT "THE NORTH WEST."
310 INPUT d
320 PRINT
330 IF tf < 40 THEN  p$ = "SNOW OR SLEET"
340 IF tf >= 40 THEN  p$ = "RAIN OR DRIZZLE"
350 IF d > 90 AND d < 250 THEN  t1$ = "INCREASING"
360 IF d > 249 THEN  t1$ = "DECREASING"
370 IF d < 91 THEN  t1$ = "NEARLY CONSTANT"
380 IF s <> 1 THEN 420
390 p1$ = "GOOD"
400 IF d < 91 THEN  p1$ = "EXCELLENT"
410 GOTO 480
420 IF s <> 2 THEN 470
430 IF d < 91 THEN  p1$ = "GOOD"
440 IF d > 90 AND d < 250 THEN  p1$ = "MODERATE"
450 IF d > 249 THEN  p1$ = "SLIGHT"
460 GOTO 480
470 p1$ = "VERY SLIGHT"
480 FOR d = 1 TO 500: NEXT d
490 HOME
```



```
500 PRINT "THE FORECAST FOR THE NEXT 12"
510 PRINT "CAST NO SHADOW, IT CAN BE"
520 PRINT "SHOWN THAT S/H IS ROUGHLY"
530 PRINT "EQUAL TO 2*3.14*D*H/C.  SO..."
540 PRINT "C=CIRCUMFERENCE=2*3.14*D*H/S"
550 PRINT
560 PRINT "WHAT IS YOUR VALUE FOR D (MILES"
570 PRINT "NORTH OR SOUTH)?"
```

Explanation

The variable definitions and the line-by-line descriptions are given here.

tf	Fahrenheit temperature
tc	Celsius temperature
s	number for sky cover
d	wind direction (degrees) and a wait counter
p$	expected precipitation
t1$	expected temperature direction
p1$	chances of precipitation

Lines 10-40. Prints the introduction.

Lines 50-100. Asks the Fahrenheit-degree temperature. Then, line 80 converts it to Celsius degrees.

Lines 120-200. Asks for the current cloud cover. If s doesn't equal one of the choices, it asks again.

Lines 210-310. Asks for the current wind direction in degrees, and explains the convention for wind direction.

Lines 330-340. Sets the precipitation type: rain for temperatures well above freezing, snow for those around or below.

Lines 350-370. Sets temperature direction based on wind direction: warmer if from the south, colder if from the north, constant if from east/west.

Lines 380-410. Sets chances of precipitation based on cloud cover (s=1=cloudy) and wind direction (d<91=north).

Lines 420-460. Sets chances of precipitation based on cloud cover (partly) and wind direction.

Line 470. Sets chances of precipitation based on cloud cover (clear).

Line 480. A wait counter to give your eyes a rest while watching the screen.

Lines 490-520. Header for the forecast.

Lines 530-570. Prints the results.

Sample Run

```
      FORECAST THE WEATHER!

WHAT IS THE FAHRENHEIT
TEMPERATURE?
?80
THAT IS 26 DEGREES CELSIUS.

ENTER THE NUMBER THAT BEST
DESCRIBES THE SKY.

1.  CLOUDY
2.  PARTLY CLOUDY
3.  CLEAR
?3

FROM WHAT DIRECTION IS THE
WIND COMING?

E=90   S=180   W=270   N=360

ENTER THE DIRECTION IN
DEGREES.  FOR EXAMPLE:  315
WOULD MEAN THE WIND WAS FROM
THE NORTH WEST.
?230

THE FORECAST FOR THE NEXT 12
HOURS IS AS FOLLOWS:

EXPECT INCREASING
TEMPERATURES.
THE PROBABILITY OF
RAIN OR DRIZZLE IS VERY SLIGHT.
```

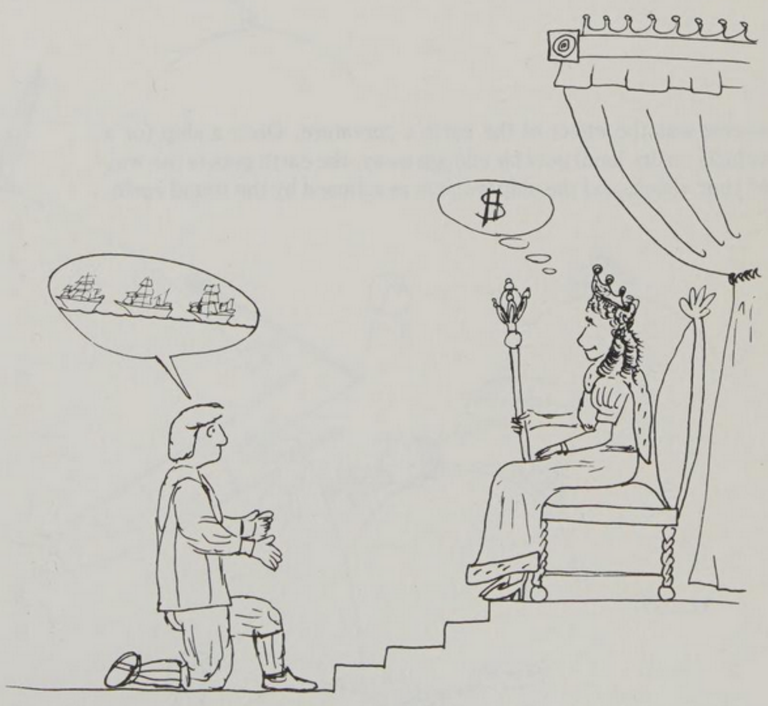
MEASURING THE EARTH'S CIRCUMFERENCE

As you may already know, people, for a long time, thought that the earth was flat. But certainly, you say, Magellan and his 'round-the-world voyage must have put an end to that kind of thinking!

Wrong. To this day, there exists a Flat Earth Society (a tradition with its beginnings in Greek mythology) whose members won't believe a word of this. So much for the power of logic.

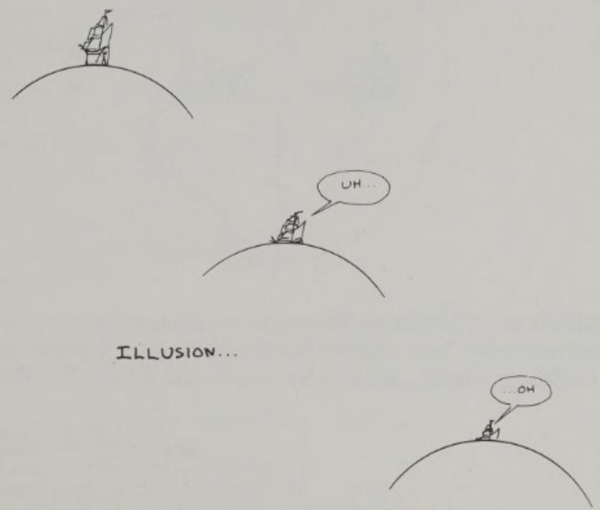
Christopher Columbus became well known for the stink he made concerning the shape of the earth. He set out to prove that the earth was a sphere and that it was indeed possible to sail a ship completely around it. In the face of many refusals by noblemen to finance such nonsense, Chris finally convinced Isabella, Queen of Spain, to pay for this experiment. In reality, he never made it

completely around the earth. He never even got to India, which was always accessible from eastern Europe. Instead, he bumped into the Caribbean Islands, and decided that he had reached the West

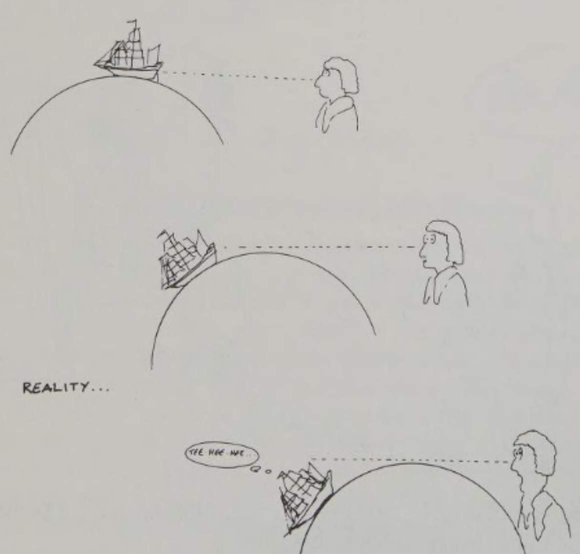


Indies. So although he never proved his original point, it's probably a good thing that he stuck to his guns.

Back to the roundness of the earth. Early sailors who ventured out into the open sea were somewhat worried that they would fall off the edge. To support these fears, was the observation from land that ships seemed to disappear over an edge. What they were really



seeing was the effect of the earth's curvature. Once a ship (or a vehicle on dry land) gets far enough away, the earth gets in the way of your vision, and the ship/vehicle is eclipsed by the round earth.

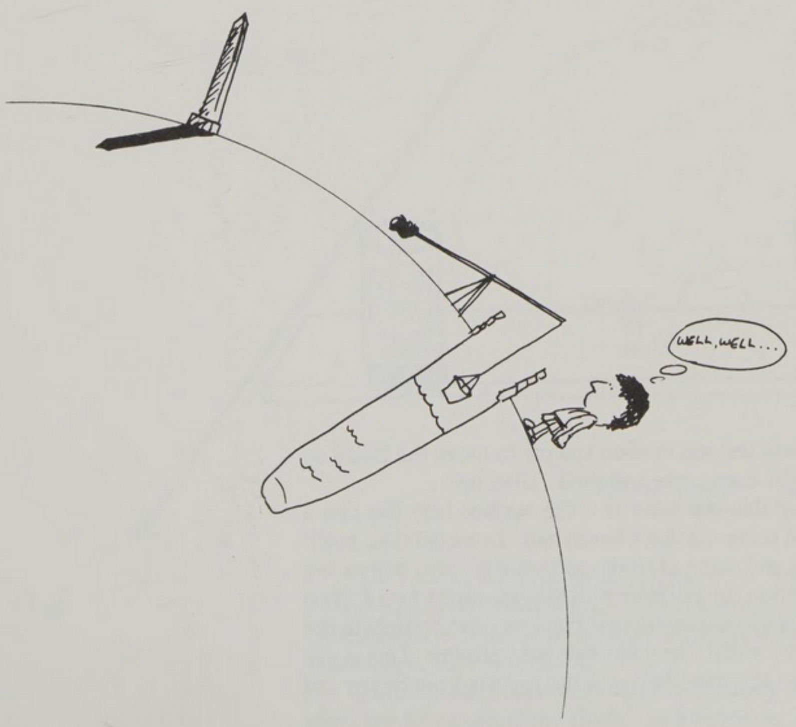


So as it turns out, our terrified landlubber ancestors were convinced of a flat earth by the very proof of its round shape.

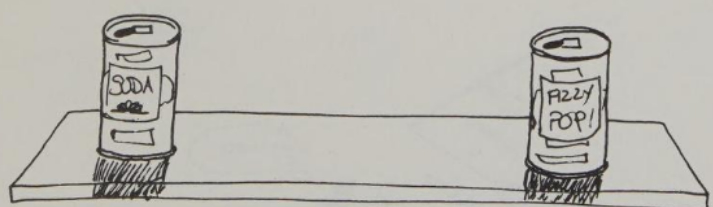
This whole problem began to clear up when sailors returned to their home ports without having seen any edge. Slowly but surely, people began to form a new model for the earth's shape that would explain the disappearances: a sphere.

Long before Columbus, in 192 B.C., a Greek by the name of Eratosthenes came up with this idea of a round earth and a way to measure its size.

Eratosthenes was intrigued by shadows which he had heard

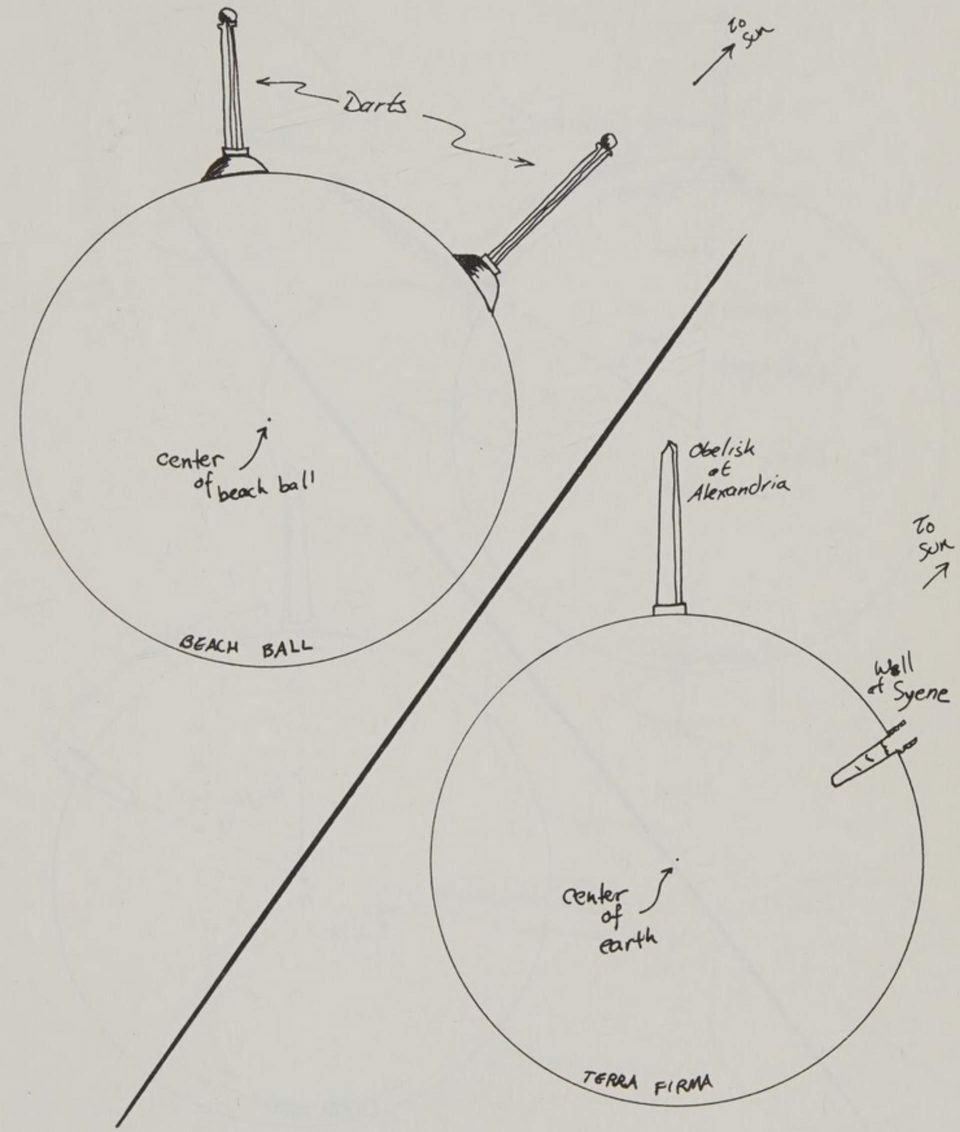


reports of. In southern Syene, Egypt (close to the equator), the noonday sun would shine straight down the shaft of a well on certain days of the year. Yet in Alexandria, 500 miles further north, there were no objects upon which the sun would shine straight down. It bothered Eratosthenes that this happened, because it couldn't be explained if the earth were flat. There simply is no way to have a vertical object cast a shadow in one place and not in another distant place, so long as both of the objects are on a flat surface. If you don't believe it, get a long board and two soda cans.



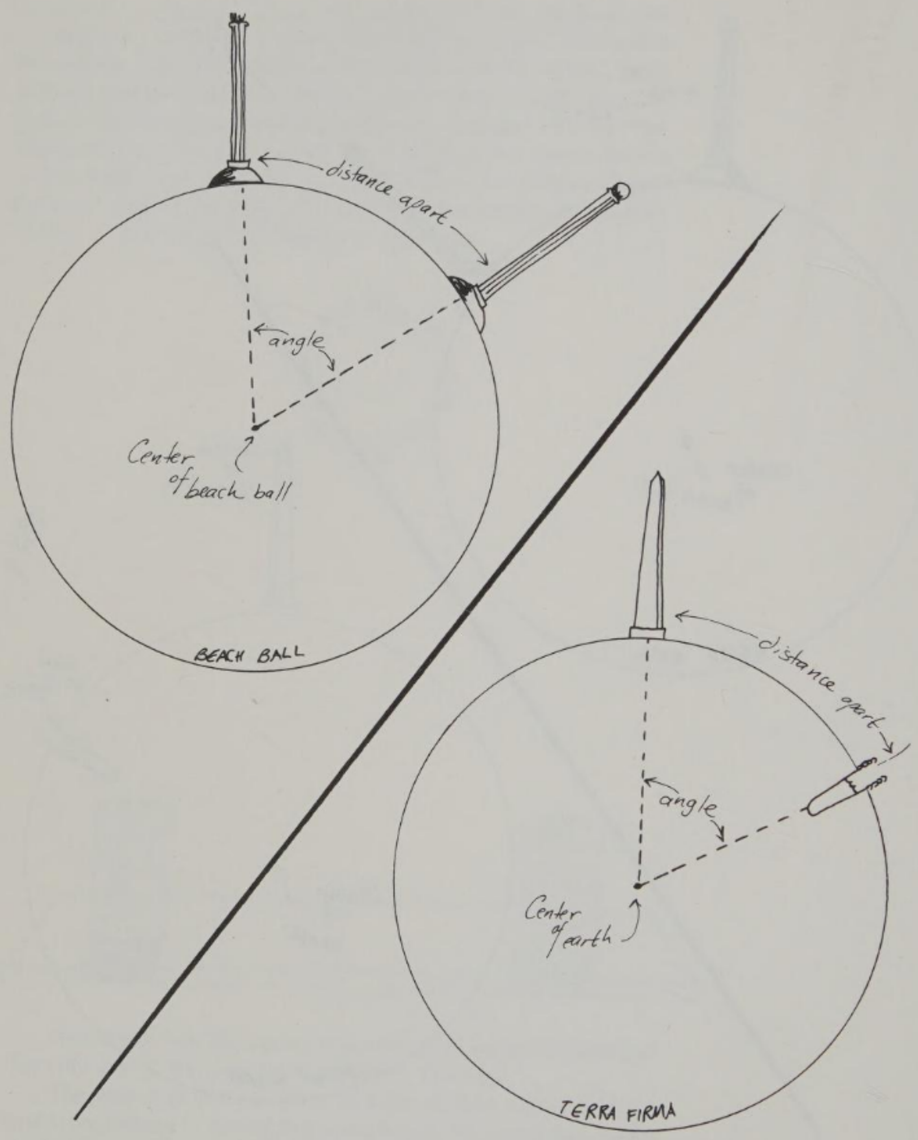
Go outside into the sun at noon and try to move the board so that only one of the cans casts a shadow. Give up?

The only way this can work is if the surface isn't flat like a board, but instead is curved like a beach ball. To model this, you'll need a beach ball and some of those suction-cup darts from a toy gun. Plant the darts on the ball instead of the cans on the board. Now you will have some success in the sun. Point one dart straight at the sun (no shadow) and viola! The other dart has a shadow. This is just what Eratosthenes observed between the reports from Syene and Alexandria.

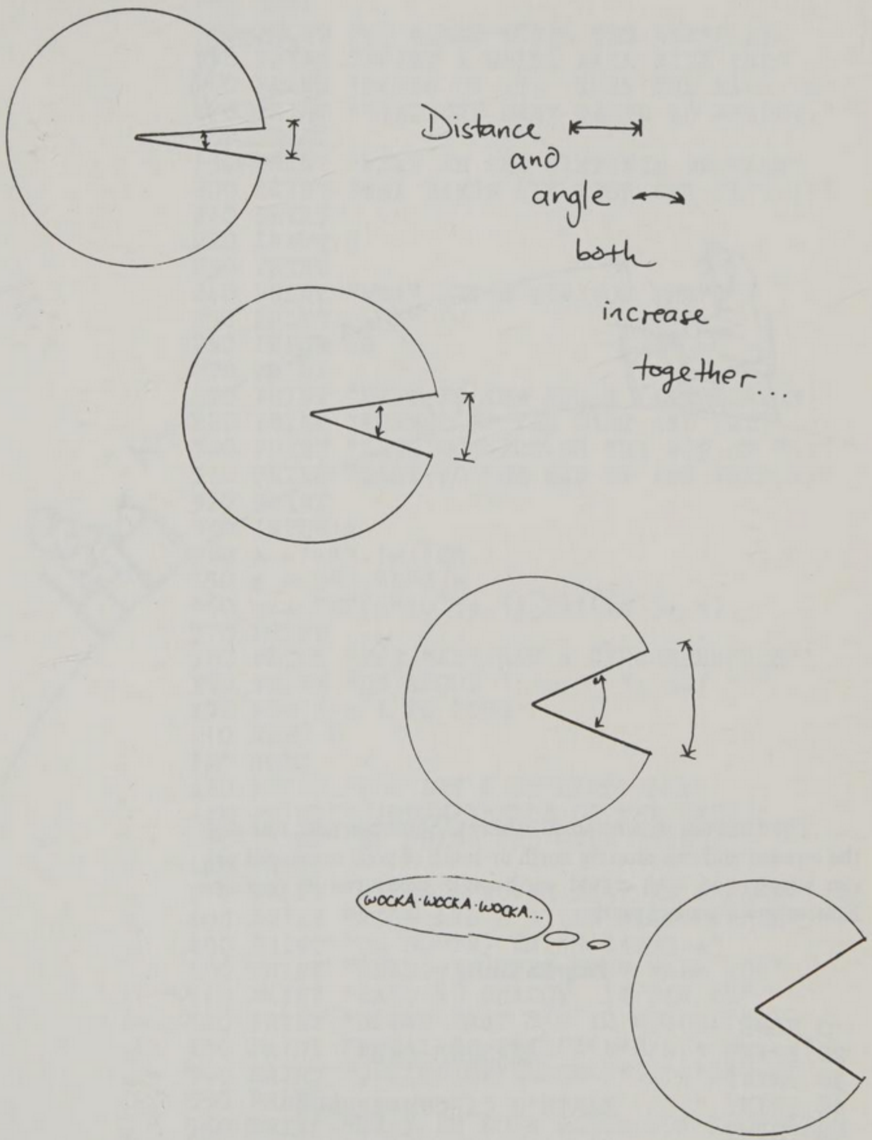


The earth must be round (like your beach ball!) if the Alexandria/Syene shadows are to be expanded. Furthermore, the two objects (your darts) must point to the center of this round earth (remember, they're vertical . . .).

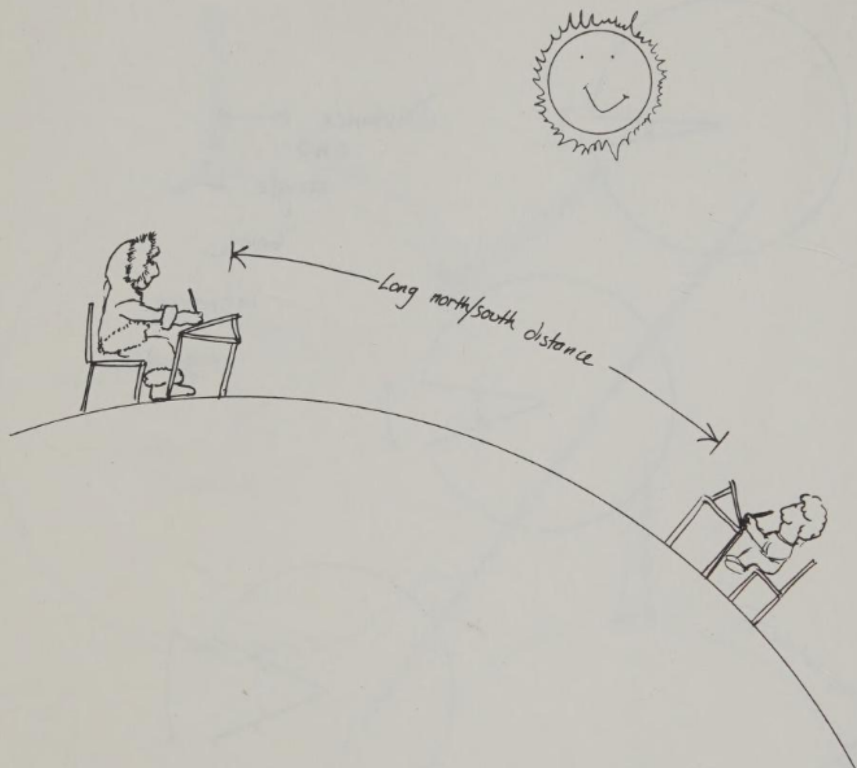
Eratosthenes knew the distance between the two sites. He also knew the angle of the shadow in Alexandria. Remembering his



geometry, he knew that the angle made by the two objects must relate to their distance apart in the same way that the total angle made by the earth's sphere relates to its circumference. Now he had it! All he had to do was to set up a proportion with these angles, and he would get a number for the circumference for the earth!



This computing is what the program does for you. You can easily use the darts on any smooth-surfaced ball. Orient the darts' shadows as we did above, and measure two things: the distance between the darts, and the angle of the shadow on the dart that has a shadow. ADAM will then tell you the circumference of the ball that you're working with.



Tired of working with small spheres? Some pen pals, one near the equator and one directly north or south of your equatorial pal, can supply you with actual earth-angle measures to recreate Eratosthenes' experiment!

Program Listing

```
10 HOME
20 PRINT "          MEASURE THE"
30 PRINT
40 PRINT "     EARTH'S CIRCUMFERENCE"
50 PRINT
60 PRINT
70 PRINT "BY USING THE MATERIALS"
80 PRINT "MENTIONED IN THE TEXT, WE"
90 PRINT "CAN MEASURE THE CIRCUMFRENCE"
100 PRINT "OF THE EARTH AND OTHER OBJECTS."
110 FOR d = 1 TO 10000
120 NEXT d
```



```
130 HOME
140 PRINT "WE FIRST MODEL THE EARTH BY"
150 PRINT "USING A SMALL BALL WITH TWO"
160 PRINT "DARTS ON IT.  TURN THE BALL SO"
170 PRINT "THAT ONE DART CASTS NO SHADOW."
180 PRINT
190 PRINT "WHAT IS THE DISTANCE BETWEEN"
200 PRINT "THE DARTS (IN YOUR OWN UNITS)?"
210 PRINT
220 INPUT d
230 PRINT
240 PRINT "WHAT UNITS DID YOU USE?"
250 PRINT
260 INPUT u$
270 PRINT
280 PRINT "WHAT IS THE ANGLE (IN DEGREES)"
290 PRINT "FORMED BY THE DART AND THE"
300 PRINT "LINE CONNECTING THE TOP OF THE"
310 PRINT "DART TO THE END OF ITS SHADOW?"
320 PRINT
330 INPUT a
340 a = a*3.14/180
350 c = 2*3.14*d/a
360 c = INT(c*10^3+.5)/INT(10^3+.5)
370 PRINT
380 PRINT "THE BALL HAS A CIRCUMFERENCE"
390 PRINT "OF ABOUT "; c; " "; u$; "."
400 FOR d = 1 TO 2000
410 NEXT d
420 HOME
430 PRINT "NOW LET'S ESTIMATE THE"
440 PRINT "CIRCUMFERENCE OF THE EARTH"
450 PRINT "BY USING THE REAL THING."
460 PRINT
470 PRINT "GIVEN A TOWER THAT IS H FEET"
480 PRINT "HIGH AND D MILES AWAY (NORTH"
490 PRINT "OR SOUTH) WHICH CASTS A"
500 PRINT "SHADOW S FEET LONG WHEN YOU"
510 PRINT "CAST NO SHADOW, IT CAN BE"
520 PRINT "SHOWN THAT S/H IS ROUGHLY"
530 PRINT "EQUAL TO 2*3.14*D*H/C.  SO..."
540 PRINT "C=CIRCUMFERENCE=2*3.14*D*H/S"
550 PRINT
560 PRINT "WHAT IS YOUR VALUE FOR D (MILES"
570 PRINT "NORTH OR SOUTH)?"
580 PRINT
590 INPUT d
600 PRINT
610 PRINT "WHAT IS YOUR VALUE FOR H (FEET)?"
620 INPUT h
```

```
630 PRINT
640 PRINT "WHAT IS YOUR VALUE FOR S"
650 PRINT " (FEET)?"
660 INPUT s
670 c = 2*3.14*d*h/s
680 c = INT(c*10^3+.5)/INT(10^3+.5)
690 PRINT
700 PRINT "THE ESTIMATED CIRCUMFERENCE"
710 PRINT "OF THE EARTH IS "; c; "."
720 END
```

Explanation

The variable list is given here. The program explanation follows the variable list.

w	delay loop counter
d	distance between objects
u$	name of units of distance
a	angle of shadow
c	circumference of sphere

Lines 10-100. Prints the introduction and explanation.

Lines 110-120. A delay loop to let you read the screen.

Lines 130-180. Background information on the method of measurement.

Lines 190-250. Gathers the number of units between darts as well as the name of those units.

Lines 260-300. Gathers the measure of the angle which was made by the dart's shadow.

Line 310. Converts a degree measure of an angle to radians, which is the way the computer deals with angle measures.

Line 320. Calculates the circumference of the sphere, based on the angle and distance.

Line 330. Rounds the circumference to three decimal places.

Lines 340-380. Displays the resulting circumference in the units which you specified (u$).

Lines 390-570. Information concerning the measurement of the actual earth's circumference.

Lines 580-600. Gathers distance (d) between you and the other object in the exercise.

Lines 610-630. Gathers the height of the tower (object).

Lines 640-660. Gathers the length of the shadow (actually measuring the angle is rather difficult).

Line 670. Calculates circumference based on height, length of shadow, and distance.

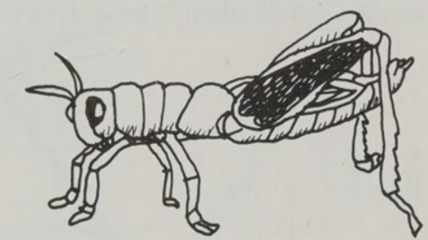
Line 680. Rounds to three decimal places.

Lines 690-720. Displays your answer.

Sample Run

```
          MEASURE THE

      EARTH'S CIRCUMFERENCE


BY USING THE MATERIALS
MENTIONED IN THE TEXT, WE
CAN MEASURE THE CIRCUMFERENCE
OF THE EARTH AND OTHER OBJECTS.
WE FIRST MODEL THE EARTH BY
USING A SMALL BALL WITH TWO
DARTS ON IT.  TURN THE BALL SO
THAT ONE DART CASTS NO SHADOW.

WHAT IS THE DISTANCE BETWEEN
THE DARTS (IN YOUR OWN UNITS)?

?23

WHAT UNITS DID YOU USE?

?cm

WHAT IS THE ANGLE (IN DEGREES)
FORMED BY THE DART AND THE
LINE CONNECTING THE TOP OF THE
DART TO THE END OF ITS SHADOW?

?30

THE BALL HAS A CIRCUMFERENCE
OF ABOUT 276 cm.
```

4



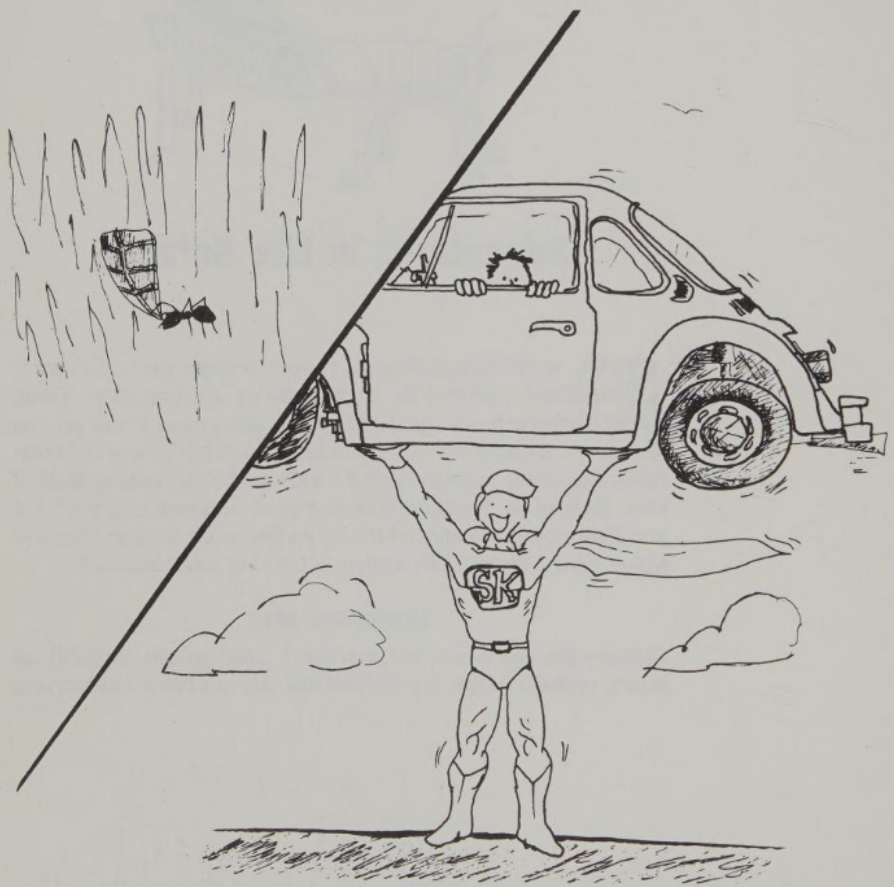
Adventures in Life Science

So far, we've toured the solar system, brought you back down to earth, and explored its characteristics and operation. While you're perfectly free to go back to those adventures whenever you please, we thought we'd take you for an in-depth look at the creatures which have managed to make a life on this fascinating chunk of rock. So read on, bold adventurer! Armed with your trusty ADAM you shall attempt to unravel the mysteries of the flora and fauna of planet earth. We're even willing to bet that you'll succeed!

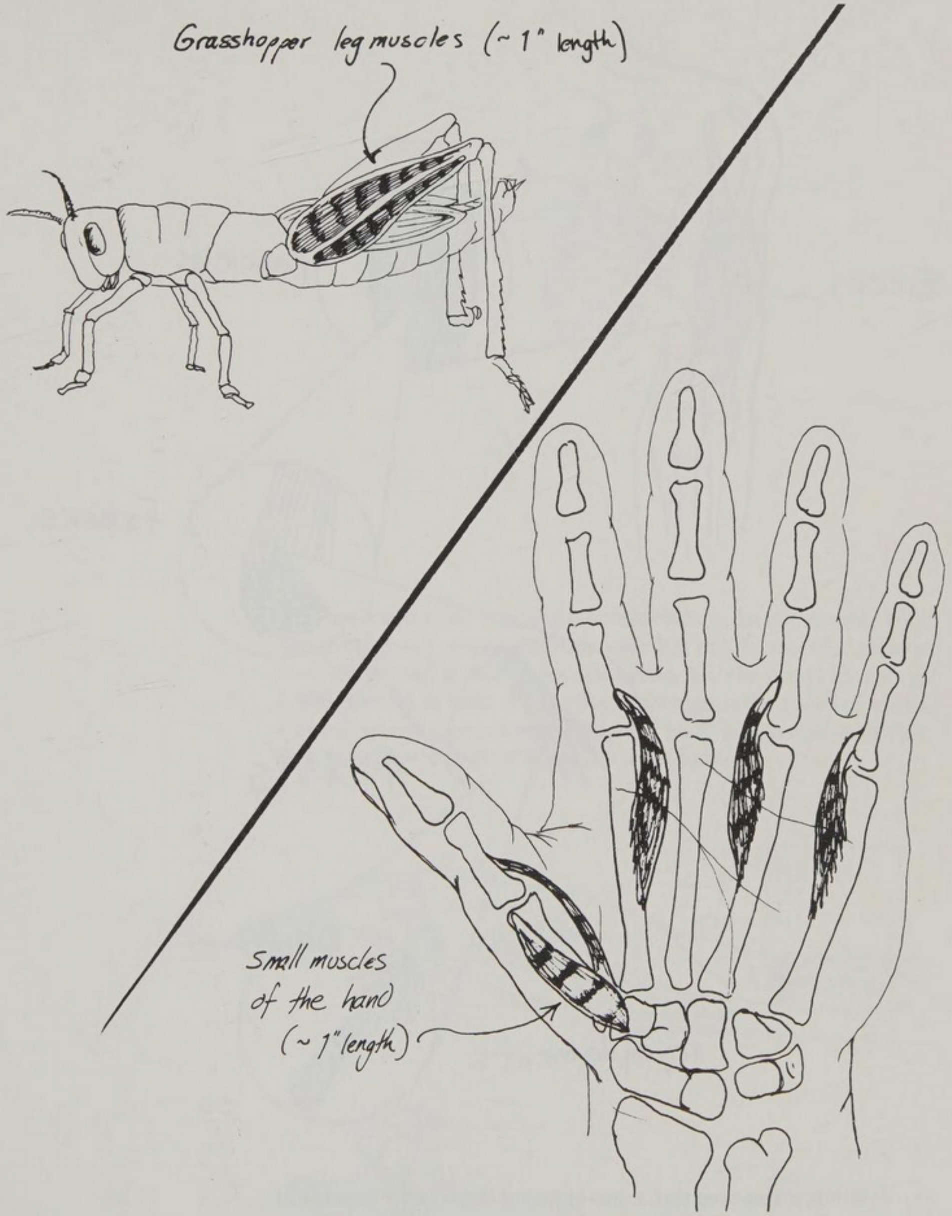
SUPERSONIC BEES

Chances are you've sat and watched a small animal, probably an insect, perhaps an ant, and realized that this tiny beast was carrying

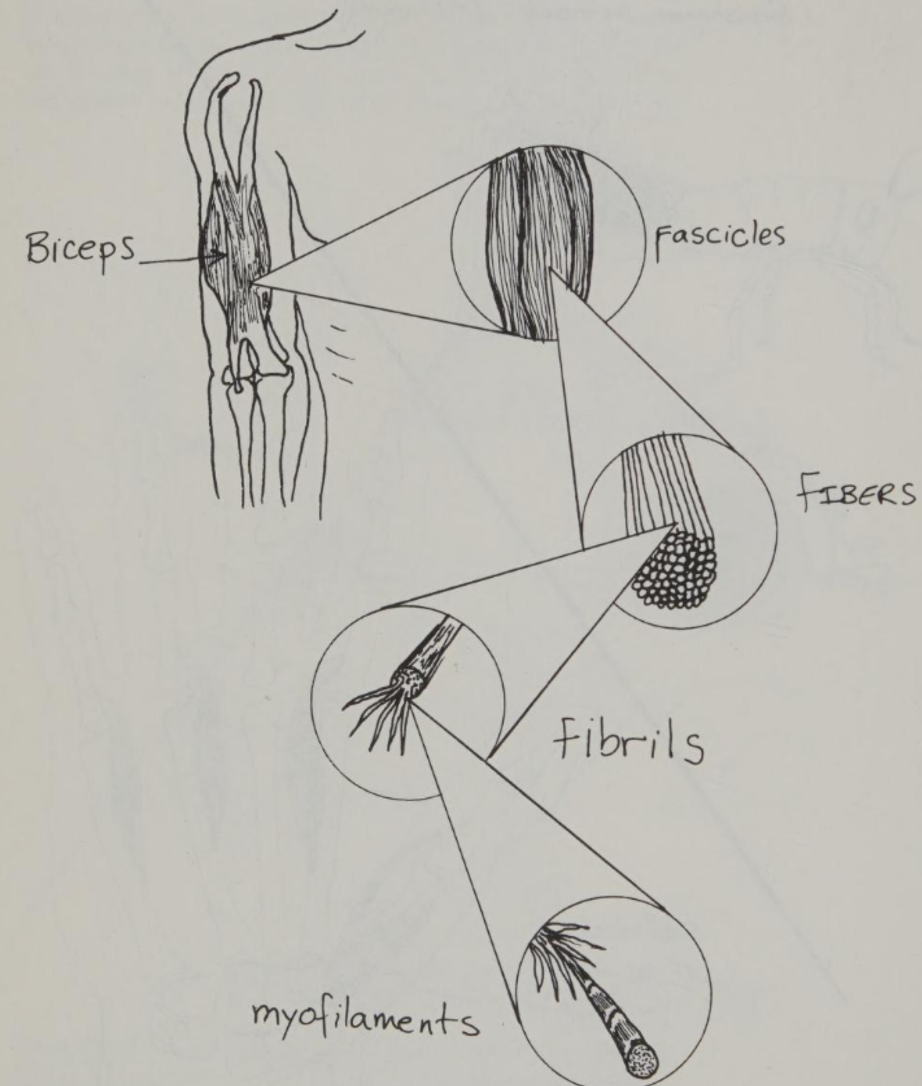
some piece of food or debris that was MANY times its own size. Maybe you've even imagined what it would be like to have performed a similar feat, and realized that to do the sort of work that the ant is doing, you would have to lift an object like a stove or refrigerator and carry it from school to home. Do ants really put



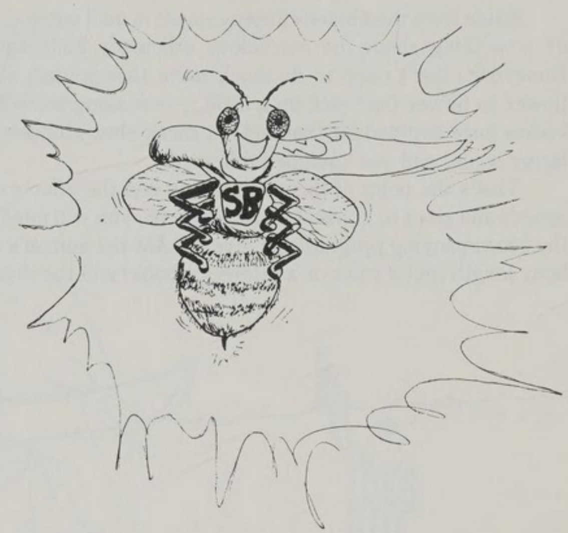
themselves through such torture? No. Insects have very small body parts. The parts are still made out of the same atoms (carbon, hydrogen, oxygen, nitrogen, and all the rest) as you are. All the parts perform in the same way, but everything is much smaller. Their largest (leg or wing) muscles, as tiny as they are, are still muscles, about as strong as your smallest (finger or toe) muscle fibers. Therefore, while an ant can't be expected to lift you, its leg muscles can be expected to do the same work as your tiny finger muscles, such as move something the size of your fingertip.



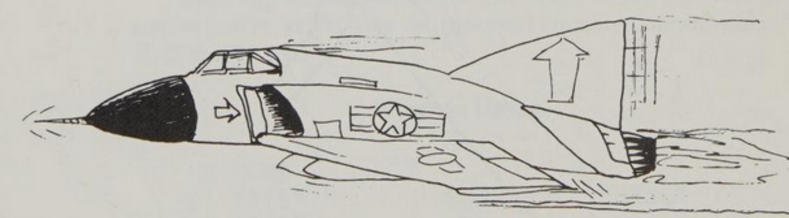
To be honest, we're the ones at a disadvantage. If we are to call on our bodies to move heavier objects, we must have many small muscle fibers working together, making for a much bigger animal.



It still remains true that a one-gram ant is perfectly capable of carrying a five-gram leaf. That's five times its own weight. And a one-centimeter honeybee can fly 8 kilometers per hour. Put in terms of its body's length, that's two million times its own length per hour. And it does all this without even breathing hard!

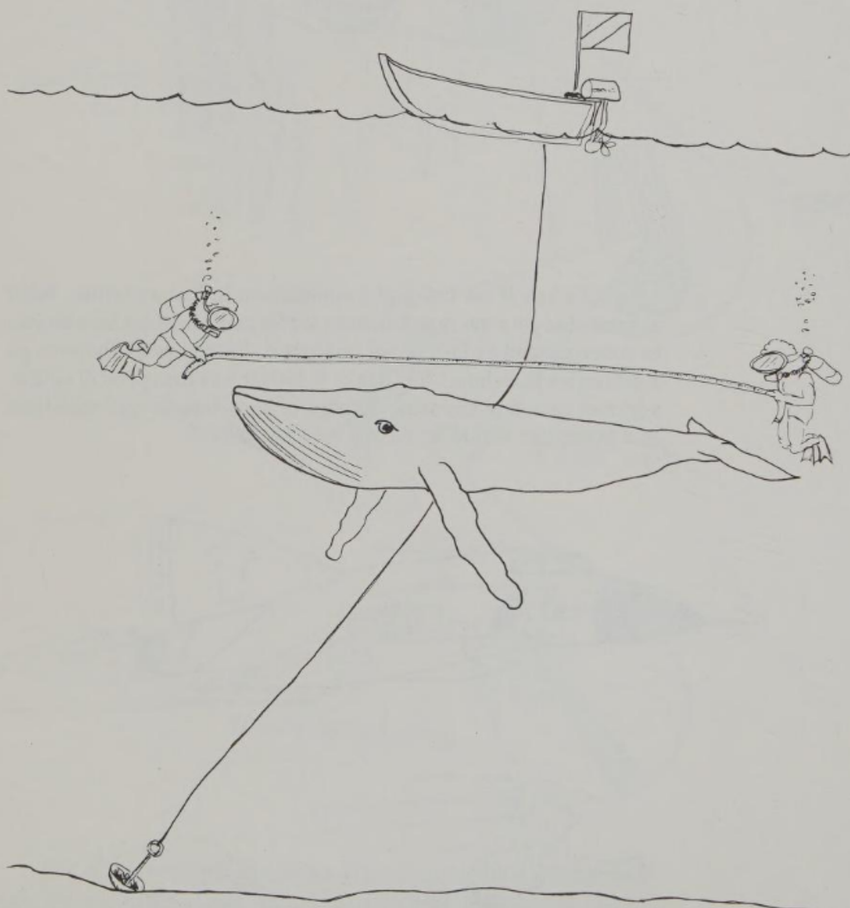


Let's see if we can find a comparison in human terms. We'll assume that your average human is six feet tall. In order for a human to cover ground as fast as our honeybee, he/she would have to go 1,409 miles in an hour! If you take 1,100 mph as the speed of sound, a human covering the same number of body-lengths-per-unit-time as a honeybee would be as fast as a jet fighter!

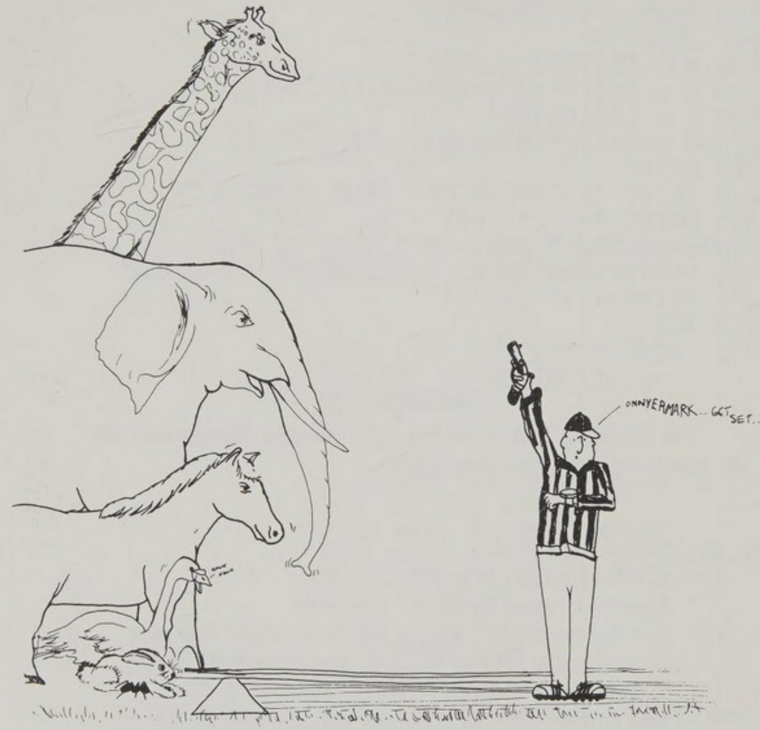


Aside from the obvious improvement in mail service, this tells us something about the marvelous efficiency built into nature. Honeybees don't need to do much more than to dart about from flower to flower (and pick up a little nectar along the way). Their bodies have evolved for that and not much else. Humans live on a larger scale, and are equipped differently.

That's the point of this program, to see the effects of scaling speeds and sizes to human terms. To do so, you will need to key in the accompanying program and give ADAM the animal's name, its body length (get a ruler or a reference book) and the distance that



the animal can cover in 15 seconds. ADAM will then tell you the actual ground-speed of that animal, and how fast it would be going if it were a 6 foot tall person.



Good luck finding out the identity of the fastest (by comparison) animal on earth! May the speediest creature (human? insect? bird? snail?) win!

Program Listing

```
 10 HOME
 20 PRINT "       SUPERSONIC BEES"
 30 PRINT
 40 PRINT
 50 PRINT "WHAT KIND OF ANIMAL ARE WE"
 60 PRINT "TESTING?"
 70 INPUT a$
 80 PRINT
 90 PRINT "WHAT IS THE BODY LENGTH OF "
100 PRINT "THE "; a$; " (IN METERS)?"
110 INPUT l
120 PRINT
130 PRINT "WHAT DISTANCE (IN METERS) DID"
140 PRINT "THE "; a$; " COVER IN"
```

```
150 PRINT "15 SECONDS?"
160 INPUT d
170 s = ((d/l)/220)*60
180 s = INT(s*10^3+.5)/INT(10^3+.5)
190 r = (d/402)*60
200 r = INT(r*10^3+.5)/INT(10^3+.5)
210 PRINT
220 PRINT "ALTHOUGH THE REAL SPEED OF"
230 PRINT "THE "; a$; " WAS ONLY"
240 PRINT r; " MPH, IF "; a$; "S WERE"
250 PRINT "OUR SIZE, THEIR SPEED WOULD BE"
260 PRINT s; " MPH!!!"
270 END
```

Explanation

The following section contains the variable descriptions and the line-by-line program description.

a$	animal name
l	a$'s body length (in meters)
d	distance covered by a$ in 15 seconds
s	distance traveled divided by 220 (body lengths per mile divided by the number of 15 seconds per minute) times 60 (minutes per hour) = scaled speed
r	real ground-speed in miles per hour (= meters per 15 sec, divided by the inverse of 15 seconds per minute times miles per meter, times minutes per hour)

Lines 10-40. Prints the introduction.

Lines 50-80. Asks the name of the animal.

Lines 90-120. Asks the body length of the animal (this must be in meters, so a one-centimeter insect is .01, a three-foot-long tortoise would be 1, etc.).

Lines 130-160. Asks the distance which you saw this animal cover in 15 seconds (so that you can use direct observations).

Lines 170-180. Converts meters per 15 seconds to a multiple of human-body-lengths-per-mile-per-minute (220), then multiplies by 60 for one hour's time. Line 180 rounds this number to three decimal places.

Lines 190-200. Converts distance-per-15-seconds to ground speed in miles per hour. Line 200 rounds to three places.

Lines 210-270. Prints these two measures, using the name of the animal.

Sample Run

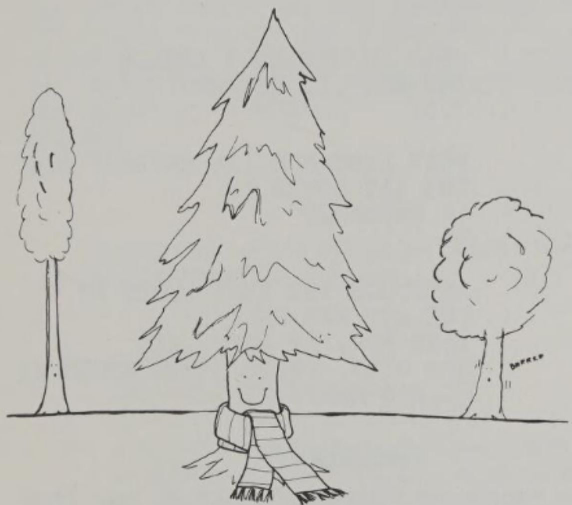
```
SUPERSONIC BEES

WHAT KIND OF ANIMAL ARE WE
TESTING?
?ANT

WHAT IS THE BODY LENGTH OF
THE ANT (IN METERS)?
?.01

WHAT DISTANCE (IN METERS) DID
THE ANT COVER IN
15 SECONDS?
?3

ALTHOUGH THE REAL SPEED OF
THE ANT WAS ONLY
.448 MPH, IF ANTS WERE
OUR SIZE, THEIR SPEED WOULD BE
81.818 MPH!!!
```

EVERGREEN KEY

Some trees lose the use of their leaves once per year. These *deciduous* trees are the ones whose leaves turn brilliant colors in the fall (where do you think that word came from) and drop off of the trees. The reason for this drastic change is that the leaves are

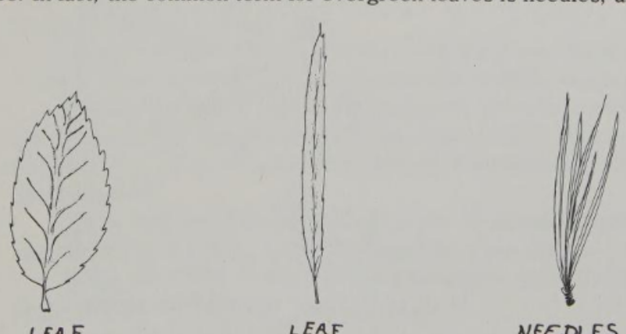
thin-walled and don't have very much sugar in them. They would never survive the winter without damaging the tree due to freezing. So nature has provided for the tree's survival in this fashion.

But there's always more than one way to skin a cat. Or in this case, to save a tree. Some trees have evolved leaves that have tough skins and high sugar contents, and do perfectly fine, all winter



long. Since these trees aren't in danger of freezing, they don't need to lose their leaves in the winter (they do lose leaves, but it happens every 2 to 14 years, depending on the species). Since they stay green all year through, they are known as *evergreens*. Pine trees are evergreens. So are spruces, hemlocks, and cedars, to name a few.

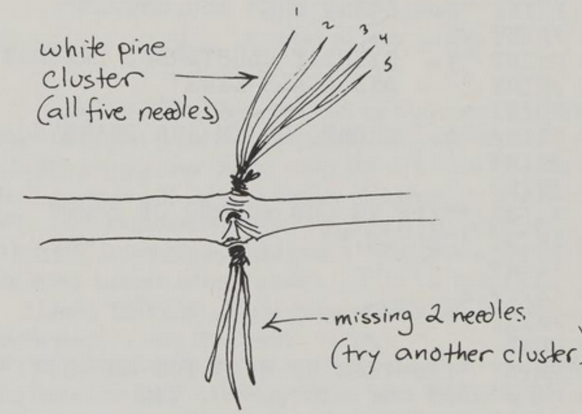
If you've ever had to deal with any of these before, you may have felt that the word "leaf" isn't the right one to use. Most evergreens feel like they have needles where the leaves ought to be. In fact, the common term for evergreen leaves is needles, and



that's how ADAM will refer to them in this program.

To unlock the identity of the various evergreens, you will need to know certain things about the needles that you've collected. You haven't collected any? Well, unless you live in the desert, there's an evergreen in the closest yard or park. That's the beauty of them. They're available all year long! The program will want to know the number, shape and size of the needles that you're investigating.

Let us mention a few notes concerning how botanists (plant scientists) deal with needles and leaves. When the program asks you to count the number of leaves in a cluster, they are referring to the number of needles that radiate from a single point of attachment.



TAMARACK BRANCH
(*Larix* species)

This in itself is an easy way to identify most of the pines, do look carefully if you think you have a pine. It is also possible that individual clusters have lost a needle or two, so use the highest number that you find in the clusters on any branch.

One last thing: if you're doing this during the spring or summer, there's one tree which may defy all attempts to identify it. It's called larch or tamarack. This tree is most common in the north, and has the maddening role of being a deciduous (leaf-losing) tree that also has needles! The drawing should help you to identify this distinctive tree.

Common Name	Scientific Name
Pine	Pinus
Cedar	Cupressis
Hemlock	Tsuga
Juniper	Juniperis
Spruce	Picea

Program Listing

```
10 HOME
20 PRINT "       EVERGREEN KEY"
30 PRINT
40 PRINT
50 PRINT "CHOOSE THE DESCRIPTION WHICH"
60 PRINT "BEST FITS YOUR TREE'S NEEDLES."
70 PRINT
80 PRINT
90 PRINT "1.  SHORT AND FLAT WITH WHITE"
100 PRINT "    UNDERSIDES"
110 PRINT
120 PRINT "2.  LONG, THIN AND SMOOTH"
130 PRINT
140 PRINT "3.  TIGHTLY CLUSTERED, PRICKLY"
150 PRINT "    AND SCALE-LIKE"
160 PRINT
170 PRINT "4.  SHORT, THICK AND SHARP "
180 PRINT
190 PRINT
200 PRINT "TYPE IN THE NUMBER OF YOUR"
210 PRINT "CHOICE."
220 PRINT
230 INPUT c
240 PRINT
250 HOME
260 PRINT "THE TYPE OF TREE YOU ARE KEYING IS A ";
270 ON c GOTO 280, 300, 510, 540
280 PRINT "HEMLOCK."
290 PRINT "INDIAN'S USED THIS TREE TO MAKE TEA.": END
300 PRINT "PINE."
310 PRINT
320 PRINT
330 PRINT "CHOOSE THE DESCRIPTION WHICH "
340 PRINT "BEST FITS THE PINE NEEDLE."
350 PRINT
360 PRINT "1.  2 NEEDLES IN A CLUSTER"
370 PRINT
380 PRINT "2.  3 NEEDLES IN A CLUSTER"
390 PRINT
400 PRINT "3.  5 NEEDLES IN A CLUSTER"
410 PRINT
420 PRINT "TYPE IN THE NUMBER OF YOUR"
430 PRINT "CHOICE."
440 PRINT
450 INPUT n
460 PRINT "THE TYPE OF PINE YOU ARE KEYING IS A ";
470 ON n GOTO 480, 490, 500
480 PRINT "RED PINE.": END
```



```
490 PRINT "PITCH PINE.": END
500 PRINT "WHITE PINE.": END
510 PRINT "CEDAR."
520 PRINT "CEDAR SHAVINGS MAKE GOOD"
530 PRINT "HAMSTER BEDDING.": END
540 PRINT "SPRUCE."
550 PRINT "SPRUCES MAKE GOOD CHRISTMAS"
560 PRINT "TREES.": END
```

Explanation

The variable list and program descriptions are given here.

c	Choice for first menu
n	Number of pine needles

Lines 10-270. These lines set up the menu of needle types which is the first set of identifiers for the tree which you are keying. Line 230 accepts your choice, while line 270 sends the computer to the line which will do some further identification. The commands ON...GOTO send control to each of the named lines (if c=1, it goes to line 280, if c=2 it goes to 300, etc.). Keep in mind that you have to put in the number of each choice, not the word or name.

Lines 280-290. These lines identify the hemlock tree, of which there is only one distinct variety.

Lines 300-500. This is a menu for the pines (you decided that you had a pine back in line 230). It too uses "ON...GOTO" to send the program control to the correct message.

Lines 510-530. Message for cedars.

Lines 540-560. Messages for spruces.

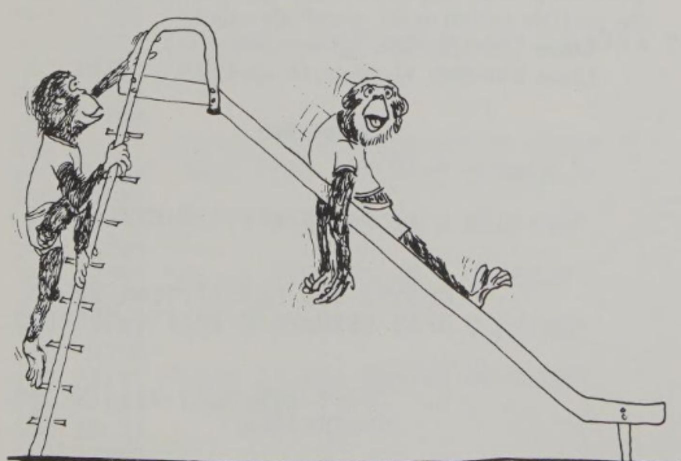
Sample Run

```
       EVERGREEN KEY


CHOOSE THE DESCRIPTION WHICH
BEST FITS YOUR TREE'S NEEDLES.


1.  SHORT AND FLAT WITH WHITE
    UNDERSIDES

2.  LONG, THIN AND SMOOTH

3.  TIGHTLY CLUSTERED, PRICKLY
    AND SCALE-LIKE
```

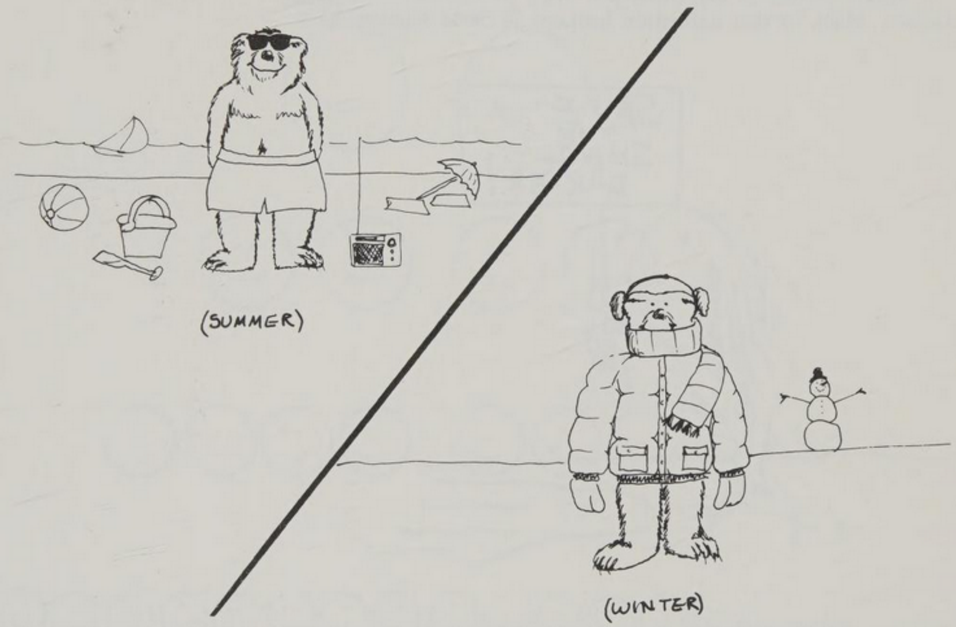
```
4.  SHORT, THICK AND SHARP

TYPE IN THE NUMBER OF YOUR
CHOICE.

?2

THE TYPE OF TREE YOU ARE KEYING IS A PINE.

CHOOSE THE DESCRIPTION WHICH
BEST FITS THE PINE NEEDLE.

1.  2 NEEDLES IN A CLUSTER

2.  3 NEEDLES IN A CLUSTER

3.  5 NEEDLES IN A CLUSTER

TYPE IN THE NUMBER OF YOUR
CHOICE.

?3
THE TYPE OF PINE YOU ARE KEYING IS A WHITE PINE.
```

POPULATION SIMULATION

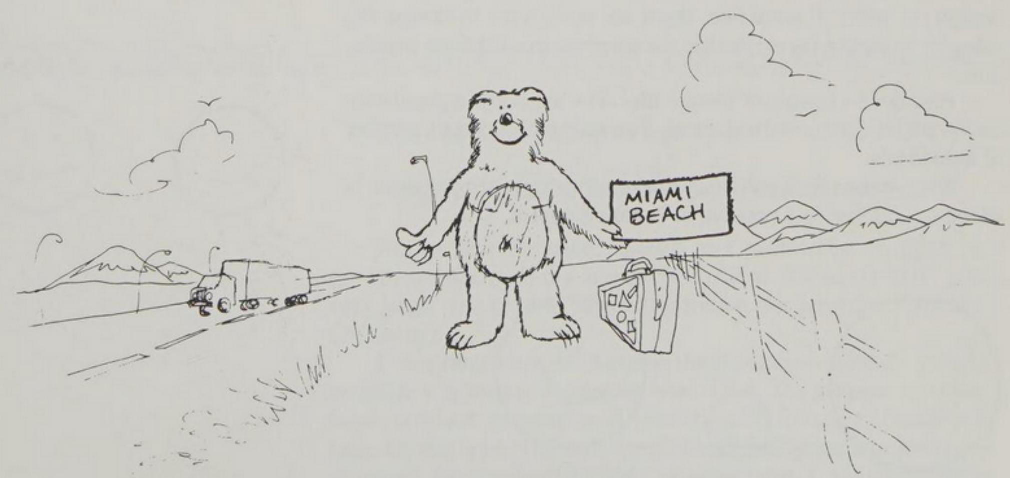
When you run around your home or school with no regard for the rules, you are often told that you're acting like an animal.



The truth is, animals behave according to rules too. Some of these rules describe how much food is allowed for each individual. Some have to do with living conditions and crowding. Others predict how many offspring each pair will have. Still others limit the temperature range of that animal, resulting in specific habitats (living places) for specific animals. The enforcer behind these laws



is nature itself. If any animal tries to overstep its bound of temperature, crowding, or feeding, death may occur. If the animal dies



before it can produce offspring, the same mistake probably won't be repeated. If this happens to all of the individuals of one type, that type of animal becomes *extinct*.

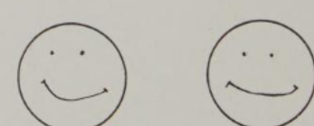
Lately, it seems that more and more animals are becoming extinct. Much of this extinction happens because humans have

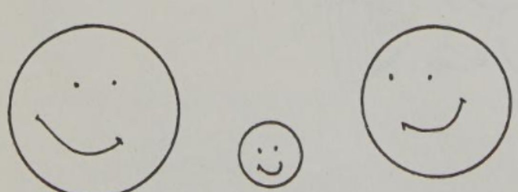
decided to change the rules by removing habitats, creating overcrowding, and removing food from the environment.

Our computer simulation is designed to show you how a population (meaning more than one individual) will develop if the rules remain the same. If you'd like, there are some ways to change the rules by changing the ways that the program predicts the population.

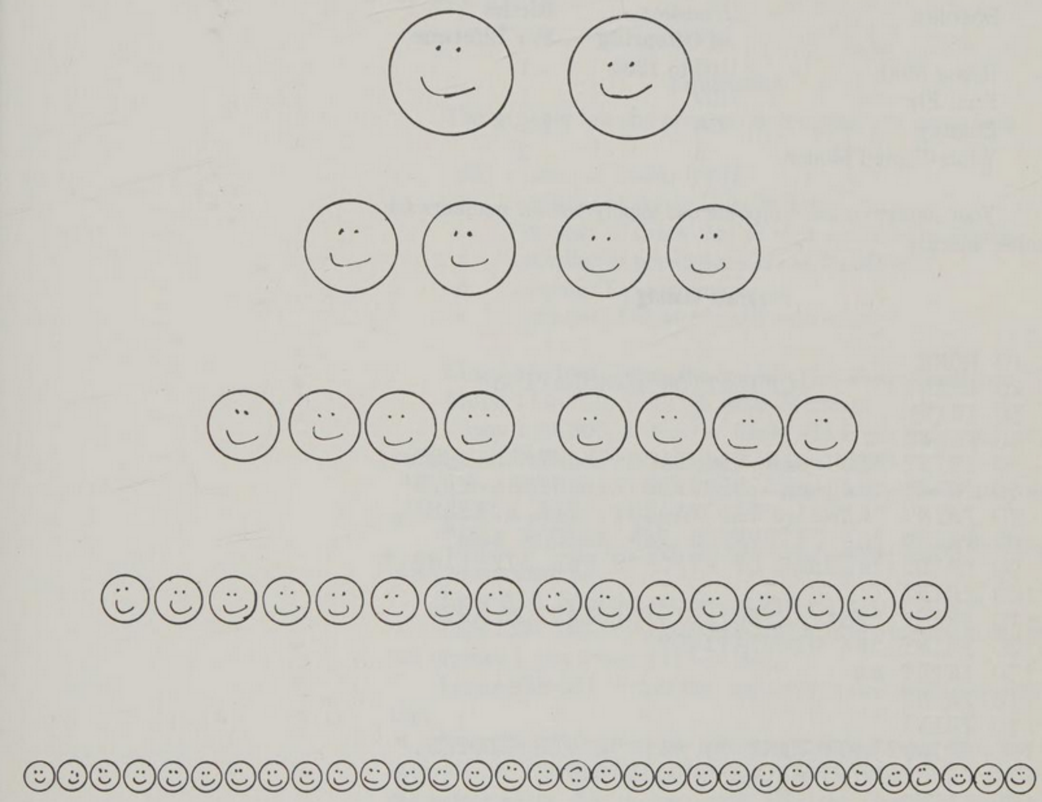
How does a computer imitate life? The history of a population can be put into mathematical terms. You start with a certain number of individuals.

Every time these individuals have offspring, the population is added to. Adding to a population is called *growth*.





Since every new member means less space and food for the population, more deaths will occur. Every time there's a death, the population gets that much smaller. Getting smaller is known as *decay*.



This can be written as an equation:

Population Size = Starting Size + Growth − Decay

Notice that we used addition and subtraction to figure in growth and decay. If the amount of births or deaths (or both) gets very large, you need to use multiplication (for growth) or division (for decay).

Every time there's a change in the population, at least one birth or death, you need to do another head count. The program does one head count per generation. A generation is defined as the time it takes for that type of animal to reach childbearing age. For each type of animal, you must tell ADAM how many times it will give birth in its lifetime, and how many offspring it has each time. From that, ADAM displays the results of each generation. Here are values for some animals to get you started:

Species	Number of Offspring	Births Per Lifetime
Gypsy Moth	100 to 1200	1
Fruit Fly	100	1
Bullfrog	250	1 to 8
White-Footed Mouse	6	3

Your library or encyclopedia can supply similar numbers for other animals.

Program Listing

```
 10 HOME
 20 PRINT "    POPULATION SIMULATION"
 30 PRINT
 40 PRINT
 50 PRINT "YOU CAN SIMULATE GROWTH RATES"
 60 PRINT "FOR ANY SPECIES OF ANIMAL WITH"
 70 PRINT "RESEARCHED FACTS:  THE NUMBER"
 80 PRINT "OF OFFSPRING PER EFFORT AND"
 90 PRINT "NUMBER OF EFFORTS PER LIFETIME."
100 PRINT
110 PRINT "WHAT SPECIES' POPULATION ARE"
120 PRINT "WE SIMULATING?"
130 INPUT s$
140 PRINT
150 PRINT
160 PRINT "HOW MANY "; s$; "S PER LITTER,"
170 PRINT "CLUTCH, BROOD, ETC.?"
180 INPUT l
190 PRINT
200 PRINT
210 PRINT "HOW MANY TIMES (NUMBER OF "
220 PRINT "LITTERS, ETC.) DO "; s$; "S"
230 PRINT "REPRODUCE IN THEIR LIFETIME?"
240 INPUT t
250 PRINT
260 PRINT
270 PRINT "HOW MANY GENERATIONS SHALL WE"
280 PRINT "RUN THE SIMULATION? (1 to 20)"
290 INPUT x
300 HOME
310 n = 2
320 FOR g = 1 TO x
330 PRINT "IN GENERATION NUMBER "; g; " THERE "
```



```
340 PRINT "ARE "; INT(n); " "; s$; "S !!!"
350 PRINT
360 n = n*((l*t)/2)
370 NEXT g
380 END
```

Explanation

The variable list and program description are given here.

s$	name of animal (plural)
l	number of young in an s$ litter
t	number of times that s$ reproduces in its lifetime
x	number of populations in the simulation
g	counter for generations
n	number of s$ after each generation

Lines 10-100. Prints the introduction and explanation.

Lines 110-150. Gets the name of the species.

Lines 160-200. Using the name which you gave, gets the number of them born every time they give birth.

Lines 210-260. Finds out how many times these animals give birth in a single lifetime.

Lines 270-300. Asks the number of generations that you want to calculate for.

Line 310. Sets the number of animals to two at first.

Line 320. The start of the loop which has g counting from 1 to the number x you entered in line 290.

Lines 330-350. Prints the number of animals in each generation.

Line 360. Calculates the number in the next generation. l × t is divided by 2, assuming that one-half of all of the animals born will die before being able to reproduce. Like the text says, you can change this for other survival rates.

Line 370. If you haven't gone through x generations yet, calculate the number of values in the next one.

Sample Run

```
    POPULATION SIMULATION


YOU CAN SIMULATE GROWTH RATES
FOR ANY SPECIES OF ANIMAL WITH
RESEARCHED FACTS:  THE NUMBER
OF OFFSPRING PER EFFORT AND
NUMBER OF EFFORTS PER LIFETIME.
```

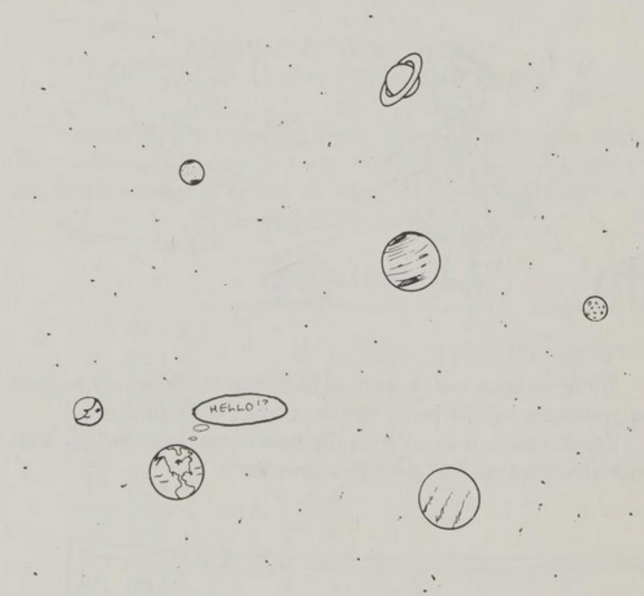
```
WHAT SPECIES' POPULATION ARE
WE SIMULATING?
?GIRAFFE


HOW MANY GIRAFFES PER LITTER,
CLUTCH, BROOD, ETC.?
?1


HOW MANY TIMES (NUMBER OF
LITTERS, ETC.) DO GIRAFFES
REPRODUCE IN THEIR LIFETIME?
?5


HOW MANY GENERATIONS SHALL WE
RUN THE SIMULATION? (1 to 20)
?5
IN GENERATION NUMBER 1 THERE
ARE 2 GIRAFFES !!!

IN GENERATION NUMBER 2 THERE
ARE 5 GIRAFFES !!!

IN GENERATION NUMBER 3 THERE
ARE 12 GIRAFFES !!!

IN GENERATION NUMBER 4 THERE
ARE 31 GIRAFFES !!!

IN GENERATION NUMBER 5 THERE
ARE 78 GIRAFFES !!!
```

BIOLOGY MATCHING QUIZ

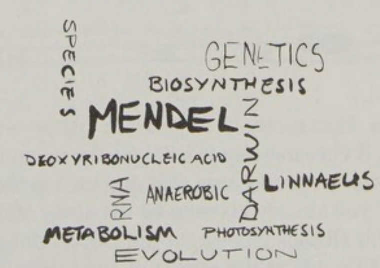
Most people seem to find biology the most exciting and intriguing of today's sciences. It's hard to pick up even the most general of magazines without seeing headlines concerning the origins of life, the debate over theories of creation, the advances in health and

medicine, and even the possibility of patenting new organisms which have been produced in laboratories.

Some of this is due to the way that biology combines and coordinates most of the other natural sciences. Some of it is due to the fact that we humans are biological creatures. Some of it is due to the fact that the inhabitants of planet earth are the only life forms



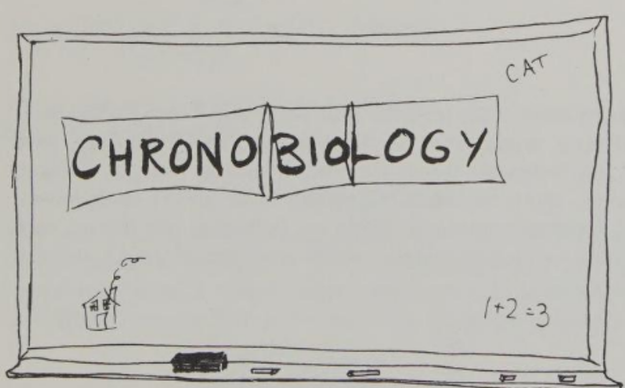
known to exist in the universe (but we're still looking). Whatever the reasons, some of the words of biology, its terminology, have crept into everyday languages. Such names as Darwin, Mendel, Linnaeus, and terms such as species, biosynthesis, metabolism, and deoxyribonucleic acid (DNA) are becoming less foreign each day. We designed this program to see how many of those terms you can recognize.



Here's a little preparatory note: The Romans had a unique way of overtaking an army that was larger than theirs. The technique was called Divide and Rule. What they would do is to isolate and overtake small segments of the enemy forces. It was much easier to outnumber ten men ten times to take on the whole hundred at once.

While we hope you never have to do that in reality, we suggest that you use a similar technique to conquer difficult terms.

For instance, if faced with the new term *chronobiology*, first divide the word up into smaller segments:



You may recognize chrono from other words, such as chronometer. A chronometer is a device for measuring (metering) time, so you can be pretty sure that chrono has the root meaning time. Biology you already know to be the study of life, so that you can now put this all back together to get chronobiology = time / life / study, or, the study of the effects of time on living things.

You may find that this helps you to understand words. Separate them according to their roots.

After running this program a few times, you'll most likely remember the answers, and the challenge will be gone. It's fairly easy, however, to change the words and definitions. They are all contained in DATA statements at the end of the program, in the following fashion:

TERM1, DEFINITION1, TERM2, DEFINITION2, TERM3, DEFINITION3, ETC. . .

Nobody will feel insulted if you decide not to stick to biology. This quiz works just as well with terms from physics, chemistry, art, literature, comic books, or whatever. It's also a great way for you and your friends to swap trivia quizzes!

Program Listing

```
10 HOME
20 PRINT "     BIOLOGY MATCHING QUIZ"
30 PRINT
40 PRINT
50 a = 1
60 b = 5
70 t = 0
80 FOR i = 1 TO a
90 FOR j = 1 TO b
100 READ a$(j), b$(j)
110 NEXT j
120 FOR k = 1 TO b
130 c = INT(RND(1)*b)+1
140 IF c$(c) <> "" THEN 130
150 c$(c) = b$(k)
160 NEXT k
170 FOR j = 1 TO b
180 PRINT a$(j); TAB(10); j; " "; c$(j)
190 NEXT j
200 PRINT
210 PRINT "TYPE IN THE NUMBER OF THE"
220 PRINT "CORRECT MATCH."
230 PRINT
240 FOR j = 1 TO b
250 PRINT a$(j); "--";
260 INPUT x
270 IF x > b THEN 250
280 e$(j) = c$(x)
290 NEXT j
300 FOR j = 1 TO b
310 IF e$(j) <> b$(j) THEN  GOSUB 400
```

```
320 NEXT j
330 IF t = 0 THEN 360
340 PRINT "YOU MATCHED "; t; " INCORRECTLY."
350 GOTO 380
360 PRINT "CONGRATULATIONS!  YOU MATCHED"
370 PRINT "THEM ALL CORRECTLY."
380 NEXT i
390 END
400 PRINT a$(j); "--", b$(j)
410 t = t+1
420 RETURN
430 DATA      DARWIN,EVOLUTION=SURVIVAL                    OF THE FITTEST
440 DATA        MENDEL,DISCOVERED THE ROLE                   OF GENES
    IN                              HEREDITY
450 DATA        FATS,NUTRIENTS USED FOR                    THE STORAGE
    OF                     ENERGY
460 DATA       D.N.A.,CHEMICAL WHICH                     CARRIES
    GENETIC                    INFORMATION
470 DATA        PROTEINS,NUTRIENT USED AS                        STRUCTURES
    AND                      ENZYMES
```

Explanation

This section contains the variable definitions and an explanation of how the program operates.

b	number of matching terms
t	number of wrong matches made
j	counter of matching terms
a$	name (item in column 1)
b$	definitions
k	counter for random printing order
c	random order for printing definitions
c$	printing assignments for b$'s
x	your answer's number
e$	your answer's definitions

Lines 10-40. Prints the header.

Lines 50-60. Sets number of terms to 5, and the number incorrect to 0.

Line 70. Begins the data-reading loop.

Line 80. Reads data from lines 400-440. The first element is the name and is called a$ (there are j of them), and the second term, the definition, is called b$ (there's j of them).

Line 90. Reads the next set of names and definitions.

Line 100. Begins the loop for picking a random order for printing the definitions on the screen.

Line 110. Picks a random number.

Line 120. Checks to see if there is already a term stored in the c$(c). If so, it picks another random number.

Line 130. Makes the Cth c$ the same as the Kth b$.

Line 140. Does this for each of the 5 terms.

Lines 150-170. Prints the terms and the scrambled definitions on the screen.

Lines 180-210. Header for asking answers.

Line 220. Begins the loop for collecting answers.

Line 230. Prints the name and a dash.

Lines 240-260. Accepts your answer, makes sure that it's 1, 2, 3, 4, or 5, then puts the term for that number into e$.

Line 270. Continues the loop for the next name.

Line 280. Starts the loop to compare answers.

Line 290. Compares e$ (your answer) to b$ (the correct answers). If they do not match, prints the correct match (lines 370-390).

Line 300. Compares the next answer.

Line 310. Checks for no wrong answers.

Line 320. Prints the number which you matched incorrectly.

Line 330. Goes to the end.

Lines 340-350. Message for no wrong answer.

Line 360. The end.

Lines 370-380. Prints the correct answers to your wrong answers. Adds 1 to t.

Line 390. Returns to line 300.

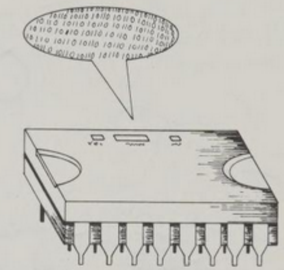
Lines 400-440. The data. On each line, there is a term, a comma, and a definition. The comma tells SmartBASIC that these are two different sets of words. Note the balloons showing the number of spaces between certain words? Those are important to make sure that the definitions appear correctly on the ADAM's screen. If you decide to change this program, like it suggests in the text, try the terms first with no spaces, then add them until the screen looks O.K. If you change the number of terms which you would like to have, change line 50 so that b equals the number of terms you want to have.

Sample Run

```
          BIOLOGY MATCHING QUIZ


DARWIN    1 NUTRIENTS USED FOR THE STORAGE OF ENERGY
MENDEL    2 NUTRIENT USED AS STRUCTURES AND ENZYMES
```

```
FATS     3 DISCOVERED THE ROLE OF GENES IN HEREDITY
D.N.A.   4 EVOLUTION=SURVIVAL OF THE FITTEST
PROTEINS 5 CHEMICAL WHICH CARRIES GENETIC INFORMATION

TYPE IN THE NUMBER OF THE
CORRECT MATCH.

DARWIN--?4
MENDEL--?3
FATS--?2
D.N.A.--?5
PROTEINS--?1
FATS--          NUTRIENTS USED FOR THE STORAGE OF ENERGY
PROTEINS--      NUTRIENT USED AS STRUCTURES AND ENZYMES
YOU MATCHED 2 INCORRECTLY.
```


5



Adventures in Mathematics and Computer Science

To get technical, this whole book to this point has been an adventure in computing. You've had to learn how to use SmartBASIC to get this far, and you've gotten used to most of the habits of your ADAM.

What we'll do in this set of adventures is take you down deeper into the workings of the computer—how it does its figuring, record keeping, etc. Since the computer is really an overgrown adding machine, we'll back things up with some practical math skills that, as scientists, we know are useful. So put away your shovel. All that you'll need to dig deeper is this book, your faithful ADAM, and some time to try the adventures that follow.

BASE CONVERTER

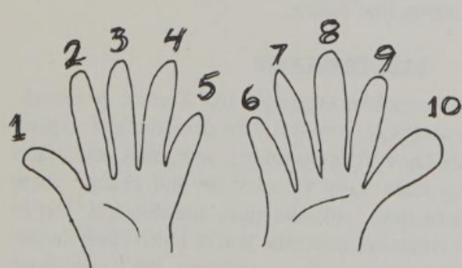
In a way, mathematics is another language, like French, or Greek, or Chinese. Like any language, you use it to describe and explain the world. You already know about numbers, numerals, and digits from math class. You know how to combine and change them through addition, subtraction, multiplication, and division. But in glancing through your computer manuals, you're more likely to see the terms *bits, bytes,* and *addresses* floating around. We'll talk about those, and how similar you and the computer are when it comes to math.

We humans usually count in *base 10*. This means that the basis for our numbering system has 10 digits (0-9). Why not 4? or 7? or 13?

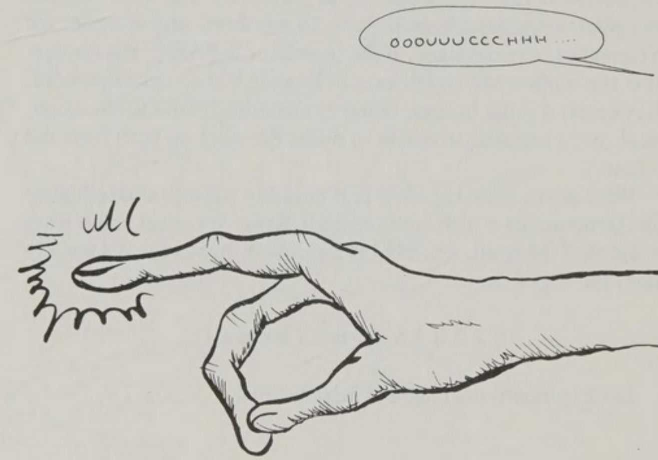
Keep in mind that we didn't always have counting machines. Some poor soul found out that he had to take care of more cattle than he could keep his eye on. He began to search for some objects that

could stand for the number of cattle he had.

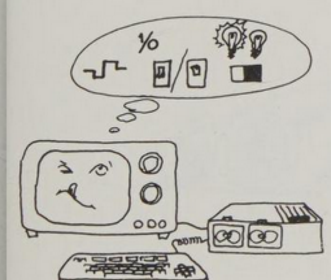
A certain genius, whose name is lost to history, found that 10 fingers made a perfectly workable counter. At that point, the sys-



tem of counting using 10 digits was born. If humans had only three fingers, we'd be counting in base three. We'd be using base 20 if we

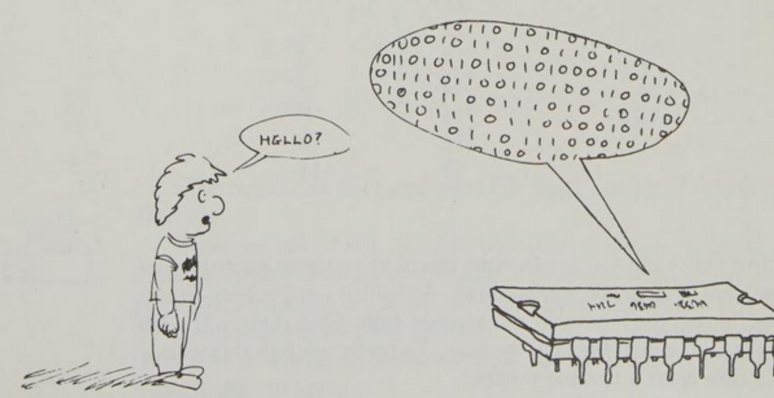


had included our toes. If that practice was ever in effect, it probably ended with the invention of shoes.

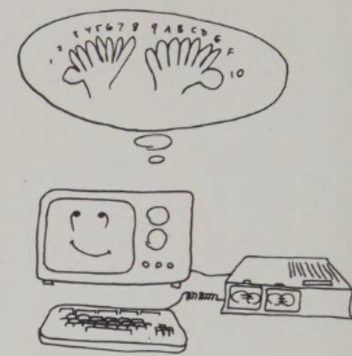


At any rate, the base that you use depends on how many digits you have to work with. Computers, being electronic, can only use base two. Why? Because the electronic switches that make them up only have two possibilities: On or Off. In numbers, this means you only have 1 or 0 to deal with (in logic, true or false, etc.). Any system which has two states is called *binary*. This is the counting system of all digital computers.

Sooner or later, humans and computers need to communicate.



This means that they must have a common language. Computers are too simple to count in decimal, and humans are too impatient to count in binary. A middle ground is struck through the use of an interpreter. If the interpreter is an *assembler*, the conversation takes place using *hexadecimal* (base 16 numbers, and is easier for the computer to understand. If the translator is BASIC, the conversation takes place in a rough form of English and decimal numbers, and is easier for the human. Other translators (FORTRAN, Logo, Pascal, etc.) are also attempts to make life easy on both man and machine.



What about base 16? How is it possible to have a base higher than the numerals which we invented? Easy. We steal some from the alphabet (after all, a symbol is a symbol). In hex (as it's usually called) the digits are:

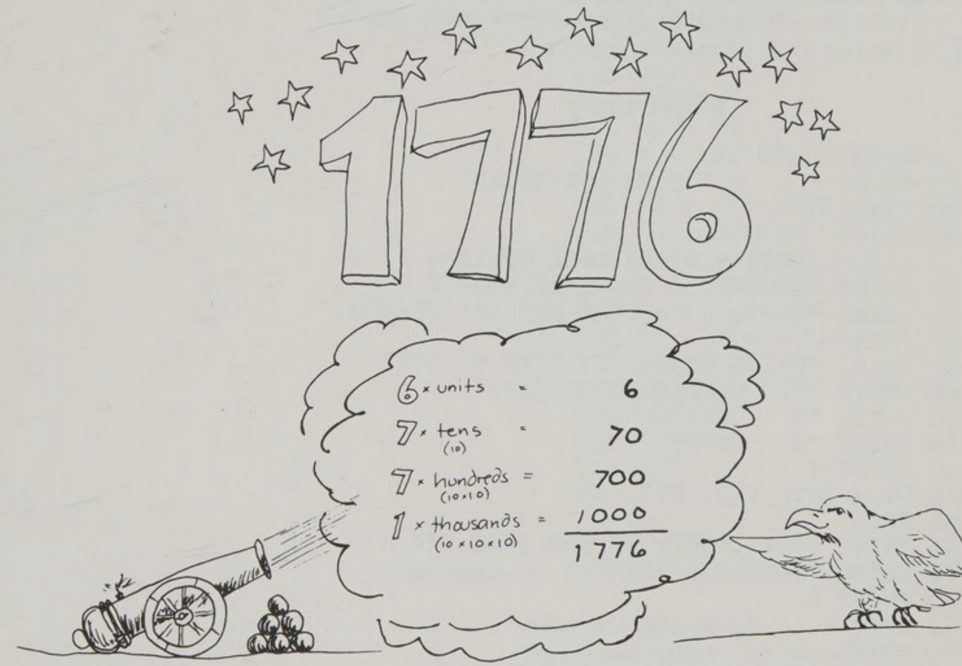
0 1 2 3 4 5 6 7 8 9 a b c d e f

Let's compare decimal and hexadecimal.

Things	Decimal	Hexadecimal
	0	0
.	1	1
. .	2	2
. . .	3	3
. . . .	4	4
. . . . .	5	5
. . . . . .	6	6
. . . .	7	7
. . . .	8	8
. . . .	9	9
. . . .	10	a
. . . .	11	b
. . . .	12	c
. . . .	13	d
. . . .	14	e
. . . .	15	f
. . . .	16	10
. . . .	17	11

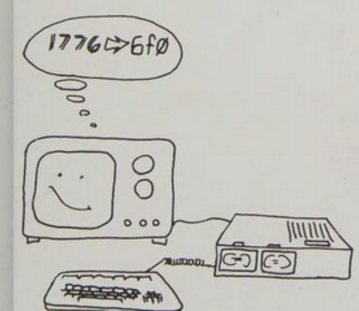
Notice that when you get as many things as you have digits in your base, you need to add another digit. And when you get base-times-base things, you'll have to add another digit. Each digit to the left stands for the next *power* of the base. To really see how this works, let's take apart a familiar number.

You can see how each numeral in the number stands for a different power of 10.



The same is true for numbers in base 16. for instance, the hex number 3e4f is really:

	f × 1	=	15
plus	4 × 16	=	64
plus	e × 16 × 16	=	3584
plus	3 × 16 × 16 × 16	=	12288
			15951



The method is the same, only the digits and place values are different.

Chances are, if you get into advanced graphics or music or other fancy programming on the ADAM, life will be much easier if you know something about hex. It's this numbering system that the ADAM uses to number the locations of all of the memory space in the machine, and all of the instructions that are needed to speed up SmartBASIC programs.

The program in this adventure will convert the two bases (decimal and hexadecimal) back and forth, depending on which one you have or need. It will also show you the same number in binary, although you have no way of speaking to your ADAM in binary.

If you'd like to test YOUR OWN conversion skills, there's a quiz at the end of the program for you to try!

Program Listing

```
 10 HOME
 20 PRINT "         BASE CONVERTER"
 30 PRINT
 40 PRINT
 50 PRINT "THIS PROGRAM WILL CONVERT A"
 60 PRINT "NUMBER OF ANY BASE FROM 2 "
 70 PRINT "THROUGH 16 TO A DECIMAL NUMBER."
 80 PRINT
 90 PRINT "WHICH WOULD YOU LIKE TO DO?"
100 PRINT "1.  SIMPLE CONVERSION"
110 PRINT "2.  QUIZ"
120 PRINT "3.  STOP"
130 INPUT q
140 ON q GOTO 150, 270, 640
150 PRINT
160 PRINT "O.K., WHAT IS THE BASE OF"
170 PRINT "YOUR NUMBER?"
180 INPUT b
190 IF b > 16 THEN 150
200 PRINT
210 PRINT "NOW TYPE IN THE NUMBER."
220 INPUT n$
230 GOSUB 390
240 PRINT "YOUR NUMBER CONVERTED TO A"
250 PRINT "DECIMAL IS "; x
260 GOTO 80
270 GOSUB 510
280 PRINT "O.K., THE BASE IS "; b
290 PRINT "THE NUMBER IS "; n$
300 PRINT "TYPE IN THE DECIMAL EQUIVALENT."
310 INPUT y
320 GOSUB 390
330 IF y = x THEN 370
340 PRINT "WRONG, THE CORRECT ANSWER"
350 PRINT "IS "; x; "."
360 GOTO 80
370 PRINT "GREAT!!!"
380 GOTO 80
390 e = 0
400 x = 0
410 FOR i = LEN(n$) TO 1 STEP -1
```



```
420 a$ = MID$(n$, i, 1)
430 IF ASC(a$) > 57 THEN 460
440 a = VAL(a$)
450 GOTO 470
460 a = ASC(a$)-55
470 x = x+a*b^e
480 e = e+1
490 NEXT i
500 RETURN
510 n$ = ""
520 b = INT(RND(1)*2)+2
530 b = 2^b
540 IF b = 4 THEN  b = 2
550 r = INT(RND(1)*4)+1
560 FOR i = 1 TO r
570 c = INT(RND(1)*b)
580 IF c > 9 THEN 610
590 n$ = n$+STR$(c)
600 GOTO 620
610 n$ = n$+STR$(CHR$(c+55))
620 NEXT i
630 RETURN
640 END
```

Explanation

The variable list and program explanation is given here.

q	choice of convert/quiz
b	base of the number to be converted.
n$	the number to be converted
x	decimal equivalent of the number in n$.
y	your quiz guess;
e	exponent for base b.
a	ASCII code for the digits in n$.
a$	individual digits in n$.
r	length of the number in n$.
c	random digit in n$

Lines 10-120. Prints the introduction and menu (convert, quiz, stop).

Lines 130-140. Asks for your menu choice, and sends control to the proper subroutine (convert=150, quiz=270, end=640).

Lines 150-190. Asks for the base of the number to be converted. If greater than 16, asks again.

Lines 200-220. Asks for the number itself (Note: if you are using base 16, hexadecimal, you must put the non-numerics A,B,C,D,E,F in CAPITAL LETTERS or the program will not work!).

Line 230. Calls the subroutine lines 390-500 (see below).

Lines 240-260. Displays your number and returns to the menu.

Line 270. Calls the subroutine (lines 510-630) which chooses a random base and number for the quiz.

Lines 280-310. Displays the base and number, and asks for its decimal equivalent.

Line 320. Calls the subroutine which converts your decimal number to the chosen base.

Line 330. Checks to see if your number matches the one which it chose, if it does, goes to the correct answer message, line 370.

Lines 340-360. Corrects you if your guess is wrong, and returns to the menu.

Lines 370-380. Prints the correct answer message and returns to the menu.

Lines 390-400. Sets the exponent (digit place minus one) and digit values to zero.

Line 410. Starts the loop which takes n$ and converts it to base b.

Line 420. Takes the last (and in the next loop, the next to the last; and so on . . .) digit in n$ and calls it a$.

Line 430. If the digit is a letter (its ASCII code is greater than 57), it goes to the line which converts to a value (line 470).

Lines 440-450. Turns the digit in a$ into a number. Goes to the converter (line 470).

Line 460. If a$ was a letter, subtracts 55 from its ASCII code to turn it into a letter.

Line 470. Builds the decimal equivalent by changing each digit to the value of that digit multiplied by the base raised to the proper exponent.

Line 480. Raises the number of the exponent (digit place minus one).

Line 490. Sends the program to line 410, lowering the digit count by one.

Line 500. Returns to line 230 or 320, depending on which this subroutine was called from.

Line 510. Clears anything that was in n$.

Lines 520-540. Chooses a random base.

Line 550. Chooses the random length of the number.

Line 560. Starts a loop which picks the random digits for the number.

Line 570. Picks a random digit. If greater than 9 (bases 11-16), goes to a conversion routine (line 610) like that in line 460.

Lines 590-600. Adds the next digit to n$, and goes to 620 to continue the loop.

Line 610. If the digit chosen was a non-numeric (A, B, C, D, E, F), adds 55 to its ASCII code to turn it into character.

Line 620. Continues the loop.

Line 630. Returns to Line 510.

Line 640. The end.

Sample Run

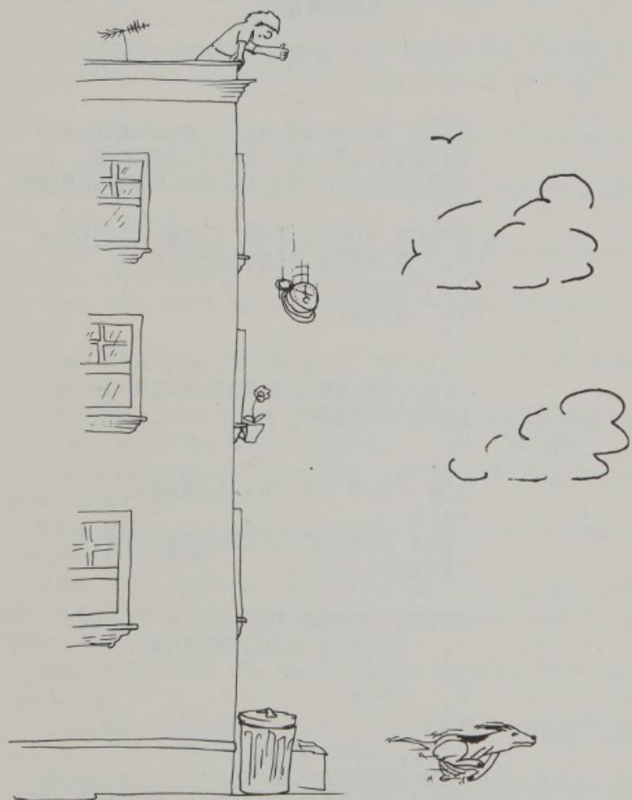
```
          BASE CONVERTER


THIS PROGRAM WILL CONVERT A
NUMBER OF ANY BASE FROM 2
THROUGH 16 TO A DECIMAL NUMBER.

WHICH WOULD YOU LIKE TO DO?
1.  SIMPLE CONVERSION
2.  QUIZ
3.  STOP
?1

O.K., WHAT IS THE BASE OF
YOUR NUMBER?
?2

NOW TYPE IN THE NUMBER.
?110
YOUR NUMBER CONVERTED TO A
DECIMAL IS 6

WHICH WOULD YOU LIKE TO DO?
1.  SIMPLE CONVERSION
2.  QUIZ
3.  STOP
?2
O.K., THE BASE IS 8
THE NUMBER IS 05
TYPE IN THE DECIMAL EQUIVALENT.
?5
GREAT!!!

WHICH WOULD YOU LIKE TO DO?
1.  SIMPLE CONVERSION
2.  QUIZ
3.  STOP
?3
```

HEIGHT FINDER

There's a famous story about a student who was taking a test in physics. One of the test questions read something like this:

"You are given a barometer and asked to use it to find the height of a tall building. How would you do it?"

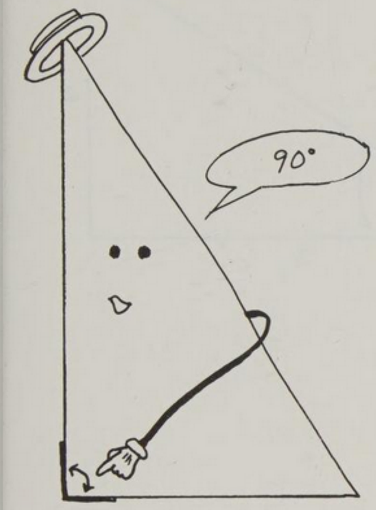
The student quickly wrote that he would go to the top of the building and drop the barometer. By measuring the time it took for



the barometer to crash to the sidewalk, he could calculate the distance to the ground.

His test was marked wrong and he was told to try again. His next answer was to tie a string to the barometer, lower it from the roof until it touched the ground, and measure the length of the string! Again his teacher marked the answer wrong and sent him back to start again.

On his third try, he suggested taking the barometer (they are fairly expensive) and offering it as a bribe if the building's custodian

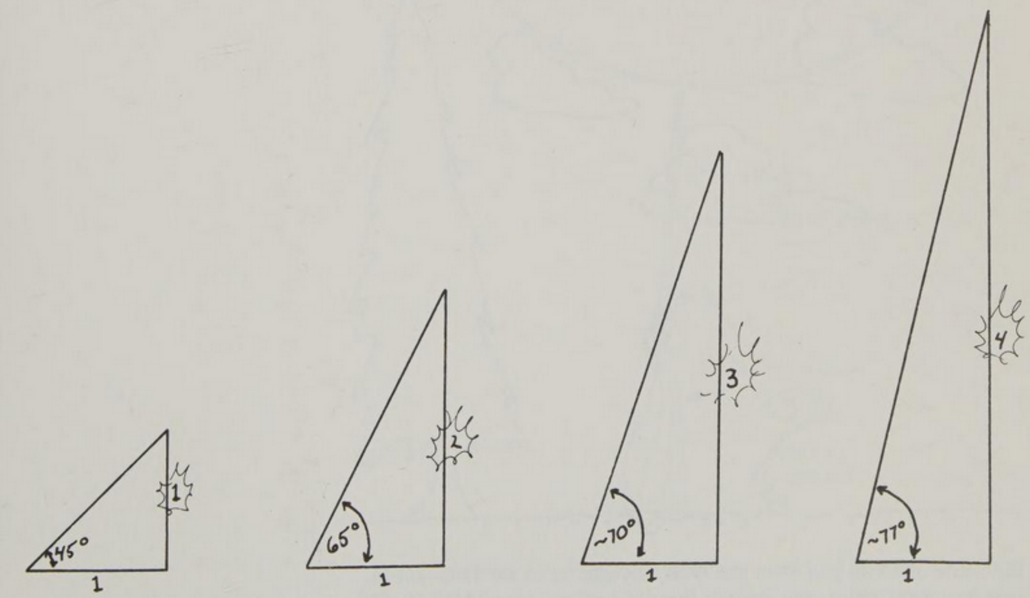


if he would tell him how tall it was. His teacher didn't like that answer either, and failed the poor student.

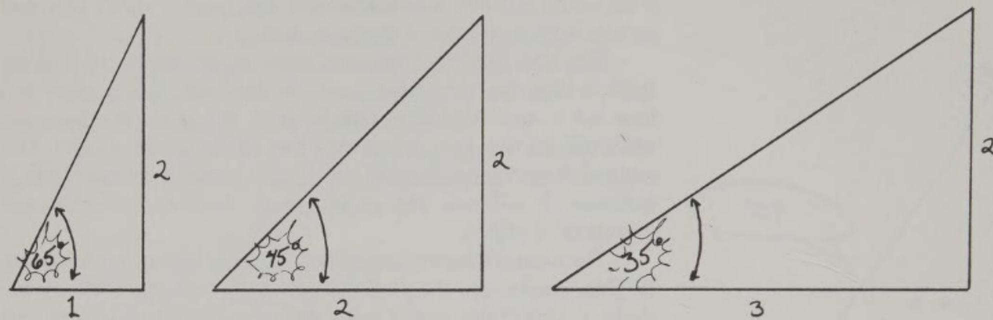
This little fable has two purposes: one, to show you that there is more than one way to solve any problem and, two, to show you how not to use ADAM to find heights. All of our methods are designed to keep your computer's feet safely on the ground. Our method of finding heights will use ADAM as an expert calculator of numbers. It will use you as an expert observer of objects and measurer of angles.

The measurement of angles is going to be the key to measuring heights. Angles are the stuff that geometry and trigonometry are made of. One of the special cases that trigonometry does very well with is ***right triangles.*** A right triangle has a right angle (90 degrees) in it. We are assuming that any object you want to find the height of is at a right angle to the ground (that means that it's standing up straight). So much for the Leaning Tower of Pisa.

One thing that a trigonometrician can tell you is that there is a constant relationship between the side of the triangle that stands up and the angle opposite it. As one gets bigger, so does the other. As one gets smaller, so does the other. "Aha!," you say, "there's a *constant* relationship between the two!" You're right. What's more,



if you decrease the measure of the angle opposite that stand-up side and leave the stand-up side the same, the base will get larger.

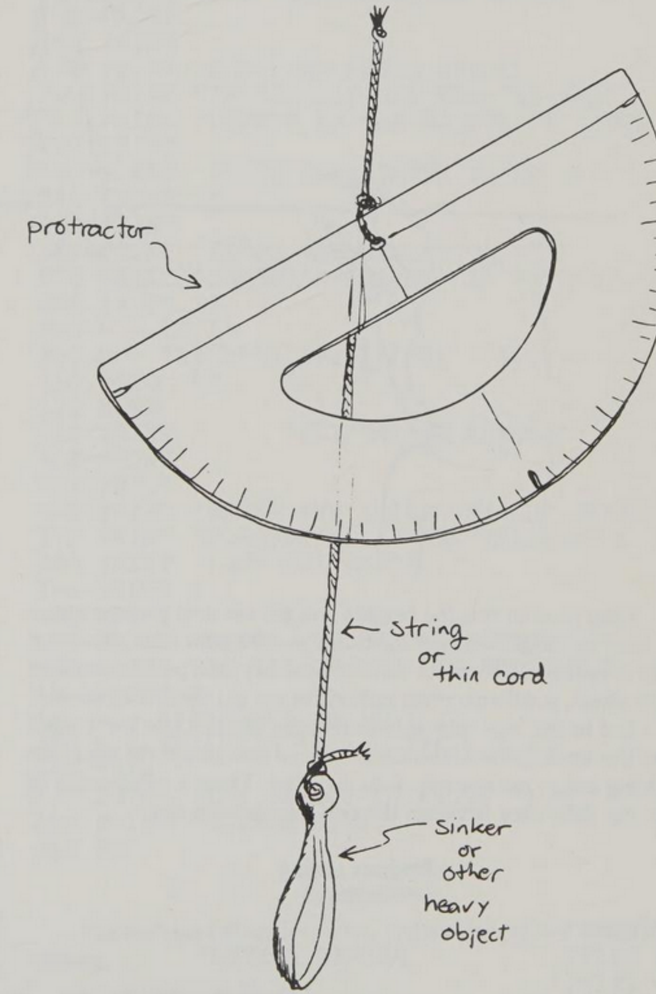


Remember that we said that the stand-up side was our object to be measured. It then means that we can call the ground our base side. The base is easier to measure. Suppose you want to measure the height of the tallest tree you know of. Step #1 is to get it in your view. Stand far enough away from it so that you can see its top without tipping your head back too far. Mark the spot where you're standing. Now measure the distance from that spot to the tree (use

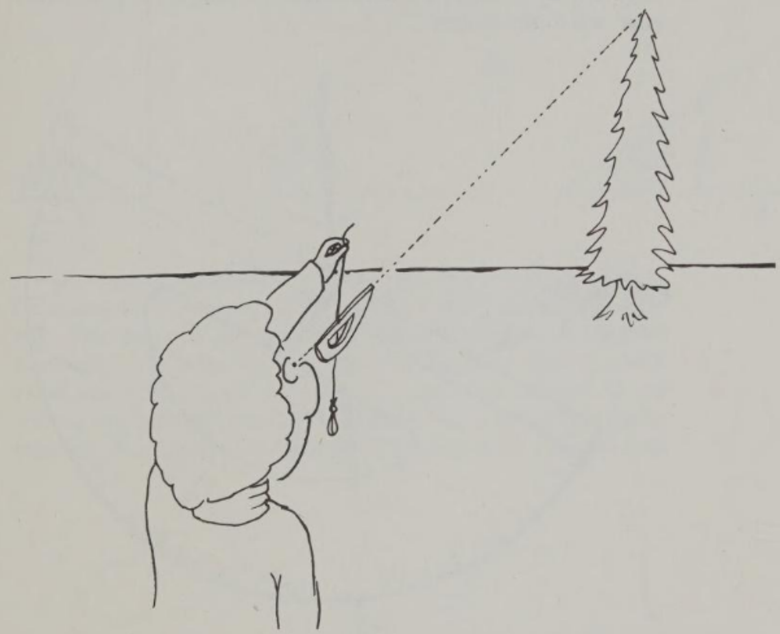
the same units as you want the tree's height to be in; feet, yards, meters, etc.). This is the number that you will call base. ADAM will ask you for this number when you use the Height Finder program.

Next you need to measure the elevation angle made when you look at the tree's top. Here's where you need some equipment.

You're going to make a clinometer. What you will need is simple: a plastic protractor (just like the one you use in school), a foot of string, and a weight (something like a large fishing sinker, a bolt, or a set of keys). Tie the string to the hole in the protractor's edge and to the weight.



Now return to that spot which you marked as your viewpoint. Hold the clinometer and line up its flat edge with the top of your tree. When the weight stops swinging, carefully trap the string against the protractor. Read the angle on the protractor where the string crosses it. This is the measure of the angle, the other number that ADAM needs. By using a set of constants called *tangents*, ADAM calculates the height of the tree (or building, or statue, or antenna) whose base and angle you measured.



Keep in mind that the heights you get are only as accurate as the base and angle that you measure, so take your time. You may want to hold some contests with friends. Let each person measure some object, and then contact authorities and get the actual number.

And by the way, the answer that the physics teacher wanted was: Use the barometer to measure the air pressure at the top of the building and at the bottom of the building. There's an equation to turn the difference between the two into feet (of air).

Program Listing

```
10 HOME
20 PRINT "        HEIGHT FINDER"
30 PRINT
40 PRINT
50 PRINT "WHICH METHOD ARE YOU USING?"
60 PRINT
70 PRINT "1.  SHADOW STICK METHOD"
80 PRINT "2.  CLINOMETER METHOD"
90 PRINT
```



```
100 PRINT
110 PRINT "ENTER THE NUMBER OF YOUR CHOICE"
120 INPUT m
130 IF m > 2 THEN 50
140 ON m GOTO 150, 320
150 HOME
160 PRINT "      SHADOW STICK METHOD"
170 PRINT
180 PRINT
190 PRINT "HOLD THE METER STICK"
200 PRINT "VERTICALLY AND MEASURE THE"
210 PRINT "LENGTH OF THE SHADOW IT CASTS."
220 PRINT
230 PRINT "ENTER THE LENGTH IN METERS."
240 INPUT l1
250 PRINT
260 PRINT "ENTER THE LENGTH OF THE"
270 PRINT "OBJECT'S SHADOW (IN METERS)."
280 INPUT l2
290 h = l2/l1
300 h = INT(h*10^3+.5)/INT(10^3+.5)
310 GOTO 460
320 HOME
330 PRINT "       CLINOMETER METHOD"
340 PRINT
350 PRINT
360 PRINT "ENTER THE ANGLE UPWARDS FROM"
370 PRINT "HORIZONTAL AS MEASURED BY"
380 PRINT "YOUR CLINOMETER."
390 INPUT a
400 PRINT
410 PRINT "ENTER THE DISTANCE FROM YOU"
420 PRINT "TO THE OBJECT (IN METERS)."
430 INPUT d
440 h = d*TAN(a*3.14/180)
450 h = INT(h*10^3+.5)/INT(10^3+.5)
460 PRINT
470 PRINT "THE OBJECT IS ABOUT "; h
480 PRINT "METERS HIGH."
490 END
```

Explanation

The variable list is given here, and the line-by-line description follows.

m	choice of methods
l1	length of meter's shadow
l2	length of shadow of object to be measured
h	ratio of shadow's lengths
a	clinometer angle
d	distance between you and the object to be measured

Lines 10-40. Prints the introduction.

Lines 50-120. Asks which method will be used.

Lines 130-140. Makes sure that you have made a valid choice, then sends you to the right routine; if m=1, it sends you to line 150, if m=2, it sends you to line 320.

Lines 150-220. Instructions for the shadow-stick method.

Lines 230-250. Asks for the meter stick's shadow length (the reference length).

Lines 260-280. Asks for the shadow length of the object being measured.

Lines 290-310. Divides 12 by 11 to get the ratio between the shadows. Because the reference was 1 meter, and the shadow angle is the same, this equals the object's height in meters. Line 300 rounds this to 3 decimal places, while line 310 sends the program to the display routine.

Lines 320-390. Asks for the angle measured using the clinometer.

Lines 400-430. Asks for the distance between you and the bottom of the object you're measuring.

Line 440-450. Calculates the height based on the distance, the tangent of the angle (the ratio between a triangle's base and its longest side . . .); a*3.14/180 is needed because your clinometer measures angles in degrees, while the computer wants angles measured in radians. There are 3.14/180 radians in each degree (or 57.28 degrees in each radian). Line 450 rounds this to 3 decimal places.

Lines 460-490. Displays the resulting heights in meters.

Sample Run

```
        HEIGHT FINDER


WHICH METHOD ARE YOU USING?

1.  SHADOW STICK METHOD
2.  CLINOMETER METHOD


ENTER THE NUMBER OF YOUR CHOICE
?1
     SHADOW STICK METHOD


HOLD THE METER STICK
VERTICALLY AND MEASURE THE
LENGTH OF THE SHADOW IT CASTS.
```



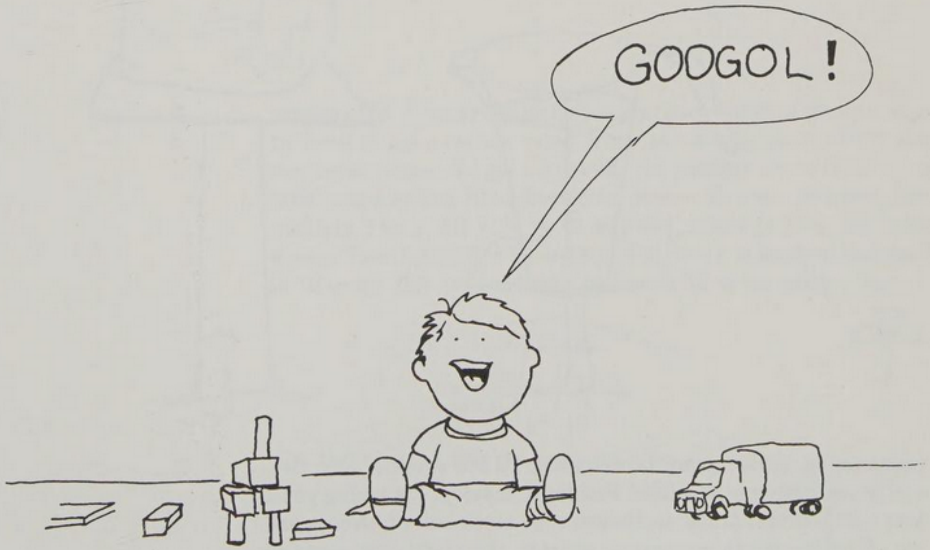
```
ENTER THE LENGTH IN METERS.
?1.5

ENTER THE LENGTH OF THE
OBJECT'S SHADOW (IN METERS).
?12.7

THE OBJECT IS ABOUT 8.46699999
METERS HIGH.
```

SCIENTIFIC NOTATION

A mathematician named Edward Kasner thinks that he knows a number so large, that nobody could ever need to use it. The number looks like this:

10,000,000,000,000,000,000,000,000,000,
000,000,000,000,000,000,000,000,000,000,000,000,000,
000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,

He knew that it was larger than a million or a billion or a trillion and that nobody had a word for a number that large. He needed to call it something, so he asked his seven-year-old nephew to give it a name. He got one.



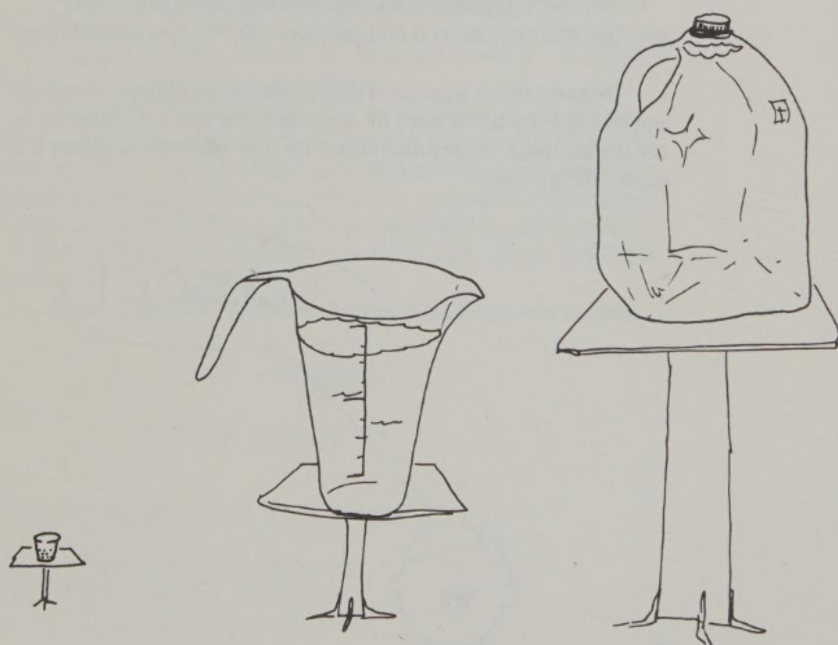
Now, *googol* isn't exactly a technical term, but surprisingly enough, the name has caught on. A physicist or astronomer can tell you that there aren't a googol of objects in the known universe.

Some mathematicians say that they have a use for the googol, but it doesn't involve counting objects.

Counting objects, though, is what most of us use numbers for. We also, as you see in our other adventures, use numbers to stand for locations, or conditions, or sizes or sometimes even words (that's how ADAM is able to deal with language). With most things, the bigger they get, the more complicated they are. This is true of numbers.

Suppose that you were in charge of all of the water in a thimble. It's your job to keep track of this water and account for it. Drops seems like a reasonable unit to use to keep track of your thimble's water. You count the drops of water in your thimble, and find out that you have 100 drops.

Your reputation as a water guard grows far and wide. Soon, you're put in charge of a pint (1920 drops) and then you find yourself

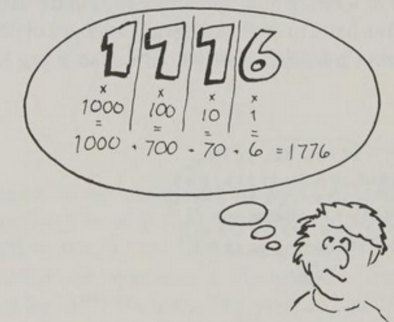


promoted to Gallon-guard (a whopping 15,360 drops!). One day, you're put in charge of a lake! Problems develop. It's taking you a long time to count drops, and besides, so much water moves in and out of the lake due to evaporation and rain, that you'll soon spend all of your time recounting.

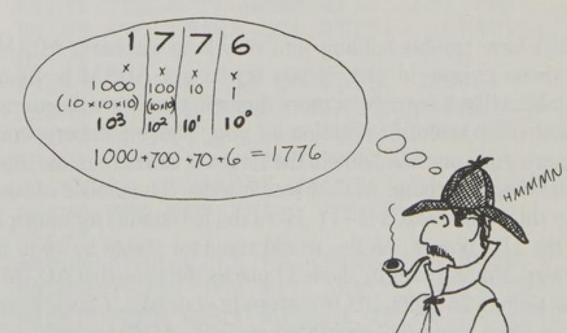
Luckily, you find refuge in *estimation*. Since being exact is almost useless, an estimate should work well enough. You start at

the geography department. They tell you that the lake has an area of 189,362 square feet, and an average depth of 12 feet, giving a volume of 2,272,344 cubic feet. Since there are 7.48 gallons per cubic foot, your lake contains 16,988,314 gallons of water. From your old guarding job, you know that there's 15,360 drops per gallon. That multiplies out to 261,094,103,040 drops! Just pronouncing that number is quite a mouthful. Most people would be quite happy if you simply told them that there were 261 billion drops in the lake. The billions are the most significant figures in this number, because most of the others (millions, thousands, hundreds, tens and units) change. There is a way to say billions in the language of numbers. Let's see how.

We humans count in base 10 (for more details, see the Base Conversion Adventure). Each of the places in our numbers stand for



a *power of 10*. A *power* equals how many times you multiply a number by itself to get a certain value. The easiest example of this is the hundreds column. To get a hundred, you multiply 2 tens (10 × 10 = 100), which equals 10 to the second power. To get a thousand, you multiply 3 tens)10 × 10 × 10 = 1000), which is 10 to the third power. Ten thousand is 10 to the fourth power, a hundred thousand is 10 to the fifth power, and a million is 10 to the sixth power.

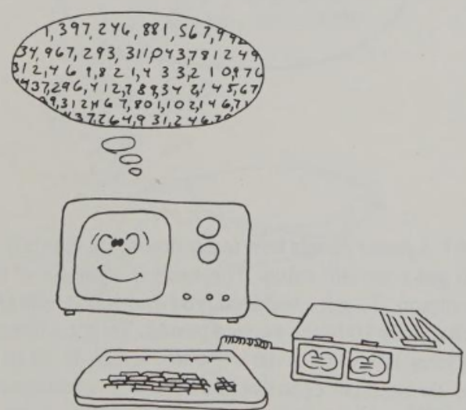


Did you pick up a pattern? The number of zeroes that a place has after the 1 is equal to its power. For the 10s place, this equals 1 (there's one zero in the number 10). How about the units? Since there are no zeroes in 1, it's equal to 10 to the zeroth power.

Let's get back to our lake. We said that it had (roughly) 261 billion drops. That's 261 times one billion. If we count the zeroes in 1 billion, we get nine. Now we can say that the number of drops is 261 times 10 to the ninth power. To put this in the language of numbers, we have to make a big jump: we will use a superscript to show that we want to raise the number 10 to a power. In the number 10^3 , the 10 is normal and the 3 is the power. That's another way of writing 1000. For a billion, it looks like this: 10^9.

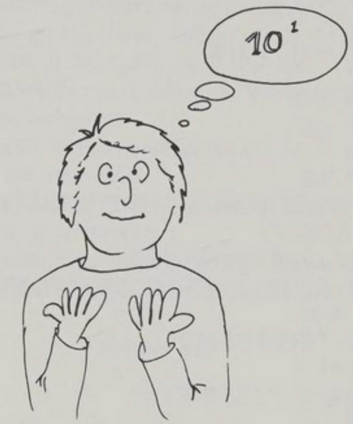
Putting all of this together for our lake, we can say that we have 261 billion, or 261×10^9.

This way of looking at large numbers is quite useful in science (where you often have to deal with large numbers of things), and has earned the name *scientific notation*. It's also a big help to brains



which have trouble holding onto very long numbers. ADAM is one of those brains. In fact, if you try to get ADAM to figure with numbers that have nine or more digits in them, it will automatically put them in scientific notation for you. These numbers won't have superscripts on them, due to the limits of a tv screen, so this is how you recognize them: ADAM would write the number of lake drops like this: 2.61094103 E+11. Here the E+ stands for multiply by 10 to the 11th power. An E– would stand for divide by 10 to the 11th power. Moving over by those 11 places tells you that ADAM figures that there's 261,093,103,000 drops in your lake. Check your figure: even when using this shorthand method, ADAM is only off by 40.

While scientific notation isn't very helpful when simply counting your fingers or toes, it can be a wonderful tool when the going gets rough and the numbers get large.



By keying-in the accompanying program, ADAM will help you to convert numbers to and from scientific notation, give you a chance to practice, and, if you'd like, quiz you on scientific notation.

So how about our large friend, the googol? Since there's 100 zeroes, it must be 10^{100}. We know that you can handle this number. See if ADAM can. If not, how large a number can it compute with?

Program Listing

```
 10 HOME
 20 PRINT "      SCIENTIFIC NOTATION"
 30 PRINT
 40 PRINT
 50 PRINT "THIS PROGRAM WILL HELP YOU"
 60 PRINT "TO LEARN SCIENTIFIC NOTATION."
 70 PRINT
 80 PRINT "SINCE THE COMPUTER CANNOT"
 90 PRINT "PRINT SMALL NUMBERS, THE"
100 PRINT "SYMBOL ^ WILL MEAN 'RAISED TO "
110 PRINT "THE...', FOR EXAMPLE:  "
120 PRINT "3.812x10^3 MEANS 3.812 RAISED"
130 PRINT "TO THE THIRD POWER."
140 PRINT
150 PRINT "YOU WILL BE GIVEN A NUMBER."
160 PRINT "YOUR JOB WILL BE TO TYPE IN"
170 PRINT "THE SAME NUMBER IN SCIENTIFIC"
180 PRINT "NOTATION.  USE NO SPACES AND"
```

```
190 PRINT "USE THE LOWER CASE x FOR "
200 PRINT "MULTIPLICATION."
210 PRINT
220 PRINT "HOW MANY WOULD YOU LIKE?"
230 INPUT m
240 c = 0
250 FOR i = 1 TO m
260 PRINT
270 GOSUB 430
290 PRINT n$
300 PRINT
310 INPUT b$
320 a$ = LEFT$(n$, 1)+"."+MID$(n$, 2, LEN(n$))+"x10^"+STR$
    (LEN(n$)-1)
330 IF b$ = a$ THEN 360
340 PRINT "SORRY, THAT IS NOT CORRECT."
350 GOTO 380
360 PRINT "GREAT!!!"
370 c = c+1
380 NEXT i
390 PRINT
400 PRINT "OUT OF "; m; " PROBLEMS, YOU"
410 PRINT "GOT "; c; " CORRECT."
420 END
430 n$ = ""
440 l = INT(RND(1)*7)+3
450 FOR j = 1 TO l
460 n = INT(RND(1)*9)
470 IF j = 1 THEN  IF n = 0 THEN  n = 1
480 n$ = n$+STR$(n)
490 NEXT j
500 RETURN
```

Explanation

The variable list and program description are given here.

m	number of trials
c	number correct
i	counter of trials
n$	number in normal format
b$	your guess for the scientific notation for n$
a$	the scientific notation version of n$
l	length of n$
n	digits to use in n$

Lines 10-210. Prints the introduction and instructions.
Lines 220-230. Asks for the number of trials you'd like.
Line 240. Sets the number correct to 0.
Line 250. Begins the loop that counts trials.

Lines 260-290. Picks and prints a random number between three and ten digits long.

Lines 300-310. Asks for your guess for the scientific notation for n$.

Lines 320-330. Converts n$ to its scientific notation, and compares it to your guess; if they match, prints the congratulatory message in line 360. If not, goes on to line 340.

Lines 340-350. Prints message for wrong answers and goes back to accept another guess.

Lines 360-380. Prints message for correct answer, adds one to c, and goes to the next trial.

Lines 390-420. If all of the trials are done, prints the number of trials and the number correct.

Lines 430-500. Subroutine to build n$ from a random number (l) of digits (n). In line 470, makes sure that the first digit is not 0.

Sample Run

```
        SCIENTIFIC NOTATION


THIS PROGRAM WILL HELP YOU
TO LEARN SCIENTIFIC NOTATION.

SINCE THE COMPUTER CANNOT
PRINT SMALL NUMBERS, THE
SYMBOL ^ WILL MEAN 'RAISED TO
THE...', FOR EXAMPLE:
3.812x10^3 MEANS 3.812 RAISED
TO THE THIRD POWER.

YOU WILL BE GIVEN A NUMBER.
YOUR JOB WILL BE TO TYPE IN
THE SAME NUMBER IN SCIENTIFIC
NOTATION.  USE NO SPACES AND
USE THE LOWER CASE x FOR
MULTIPLICATION.

HOW MANY WOULD YOU LIKE?
?2

30665161

?3.066515x10^6
SORRY, THAT IS NOT CORRECT.

327177605
?3.27177605x10^8
GREAT!!!
```

```
OUT OF 2 PROBLEMS, YOU
GOT 1 CORRECT.
```

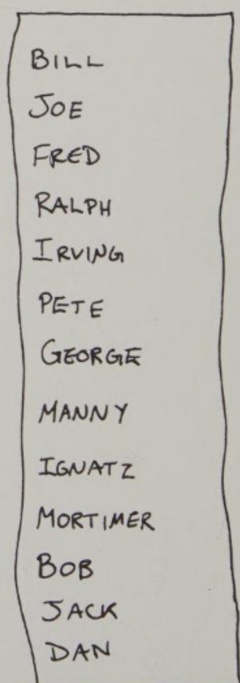
ALPHABETIC ORDER

Computers are wonderful for doing all of the things that would either take a human brain too long to do or would be too boring for a human mind. Solving equations, doing many math problems, arranging information, and landing a machine that can't really fly in the

first place (space shuttle) are all best left to computers. We're going to focus on arranging information.

Suppose that you've decided to use ADAM to keep a list of all of your friends' addresses, or the list of references for your term paper, or the words that you've had to learn in biology class. Finding a single entry in this list when you need it could be maddening if the list is a jumble, with no rhyme or reason to its order. What your list needs is some hierarchy, something that will never change, even if the list gets bigger or smaller.

There is something in our language which is stable enough to help you out. While words, their spelling, and even their pronunciation all change over time, the list of letters that make up these words hasn't changed for thousands of years. That list is called an *alphabet* (the name comes from a very old letter-list, the Greek, in which Alpha and Beta were the leading letters. Now all letter-lists are called alphabets). Almost all word lists are organized by letter-lists (alphabets). Looking for bagels is not easy in this list:



BUNDLES
BEAGLES

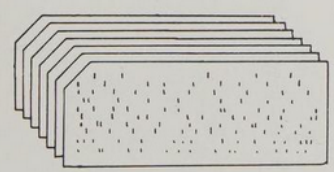
BARGAINS
BUGLES
BUNGLES
BANGLES
BAGELS
BARGES
BOODLES

Putting the words in order of their letters makes life easier:

BAGELS
BANGLES
BARGAINS
BARGES
BEAGLES
BOODLES
BUGLES
BUNDLES
BUNGLES

The method used here is to arrange them according to the order of the alphabet, starting with the first letter of each word. Since all of our words had the same first letter (they all tied for first), we had to use the second letter. If the second letters are a tie (like BAGELS, BANGLES, BARGAINS, and BARGES), you use the third letter, and so on.

This is simple enough for a human if the list is short. But as the list gets longer, it takes more time to scan and arrange, and the chance of ties in more places gets higher with each new entry.



This list-keeping, by the way, is what started the development of computer technology. The U.S. Census Bureau predicted that by the 1890 census, it would take 12 years to list and tally all of the people in the country, using paper and pencil. The trouble was, they were supposed to do a census every 10 years. The job of developing a computing machine was given to Herman Hollerith, who improved upon the machines used to automate textile mills. The census went on as planned, and to this day, computer users keep information on Hollerith-type punchcards.

Enough trivia. The program for this adventure will organize lists for you. The list must be kept in a special part of the program, called DATA statements. You can add more DATA statements as the list needs to get bigger, but remember that each numbered data line can have no more than 256 items (names, addresses, whatever) and that each element in the list is followed by a comma.

You might want to keep this program handy during holidays, to keep track of gifts. You will also use it a great deal if you need to write a bibliography for school.

Program Listing

```
 10 HOME
 20 PRINT "       ALPHABETIZE WORDS"
 30 PRINT
 40 PRINT
 50 PRINT "ENTER THE WORDS TO BE SORTED."
 60 PRINT "PRESS RETURN ONCE AFTER YOU"
 70 PRINT "ENTER EACH WORD.  PRESS RETURN"
 80 PRINT "TWICE AFTER THE LAST WORD"
 90 PRINT "ENTERED."
100 PRINT
110 DIM w$(100)
120 n = 1
130 INPUT w$(n)
140 IF LEN(w$(n)) <> 0 THEN  n = n+1: GOTO 130
150 FOR i = 1 TO n-1
160 s = 0
170 FOR j = 1 TO n-1
180 IF w$(j+1) >= w$(j) THEN 230
190 v$ = w$(j+1)
200 w$(j+1) = w$(j)
210 w$(j) = v$
220 s = 1
230 NEXT j
240 IF s = 0 THEN 260
250 NEXT i
260 PRINT
270 FOR i = 1 TO n
280 PRINT w$(i)
290 NEXT i
300 END
```

Explanation

The variable list and program description are given here.

w$	the words to be sorted
n	the number of each word (first, second, etc.)
i	rank (alphabetical) of each word
j	number of times that comparisons are made between words
s	a toggle counter to keep track of the number of comparisons made
v$	current highest ranking word in list

Lines 10-100. Prints the introduction and instructions.

Lines 110-120. Sets values: DIM means reserve enough space for 100 w$s (100 words); n, the counter for the original order of words equals 1 the first time.

Line 130. Accepts a word and calls it w$ number n.

Line 140. If you typed only a return then it means that your list is done, and the program goes to the ordering routine (line 150); if not, raises the n counter and goes to line 130 again.

Line 150. Starts the loop of words to be compared.

Line 160. Sets the toggle counter (s) to zero (meaning that there are still some words which may have a higher rank than the one being tested).

Line 170. Starts comparing one word to all others.

Line 180. Checks to see if the next word in the list is higher than it. If so, skips all the rest, and uses the next word to compare.

Lines 190-210. If no next word is higher, it stores this word in v$, and moves the next word to be tested into w$(j).

Line 220. Sets the toggle s to 1, which means that there are more words that need to be checked.

Line 230. Sends the loop back to compare the next word(s).

Line 240. If there are no more words to check, sends the program to the display section (line 260).

Line 250. If there are words left to check, sends the loop back to line 150 to continue checking.

Lines 260-300. Prints the words out in alphabetical order.

Sample Run

```
       ALPHABETIZE WORDS


ENTER THE WORDS TO BE SORTED.
PRESS RETURN ONCE AFTER YOU
ENTER EACH WORD.  PRESS RETURN
TWICE AFTER THE LAST WORD
ENTERED.

?DOG
?ADAM
?ZEBRA
?COMPUTER
?CAT
?BASIC
?LION
?
```

```
ADAM
BASIC
CAT
COMPUTER
DOG
LION
ZEBRA
```

PRETEND DATA

So far, our adventures have brought you through outer space, the atmosphere, into volcanoes and riverbeds, and into your own backyard to collect data. What this adventure will do is test your new-found (or re-awakened) skills in handling data and making conclusions based on that data.

Most people have a mental picture of scientists. Some of these are a little off-base. People often assume that scientists spend their time in dimly lit laboratories, hovering over smelly, mysterious experiments that nobody ever hears about anyway.

That image is partly imagination and partly the stereotype that we see in tv, films, and cartoons. In reality, a scientist is not unlike a detective. A scientist usually chooses a topic to his or her liking and

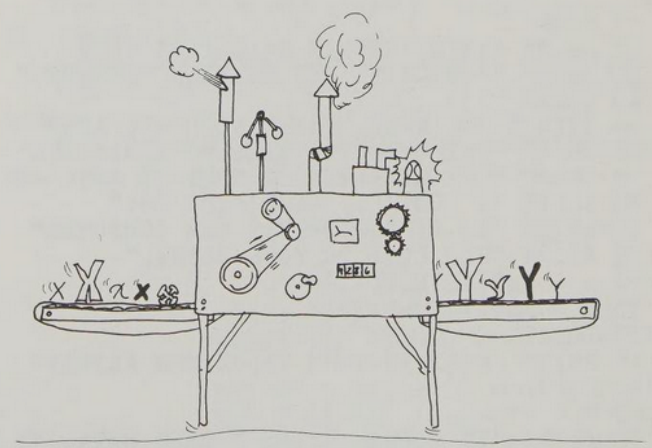
proceeds to spend his or her life (or a part of it) uncovering new information about that topic. The end product of the vast majority of science is for the improvement of life here on earth. This applies for most of the space program as well as medicine, chemistry, and engineering. Scientists work to better understand nature and to

help us use that understanding to make a better world. The results often depend on the person(s) asking for the scientists' help. John Kemeny, one of the men who invented the BASIC computer language, compares the role of the scientist to the Cheshire cat in Alice in Wonderland. When Alice asks, "Would you tell me, please, which

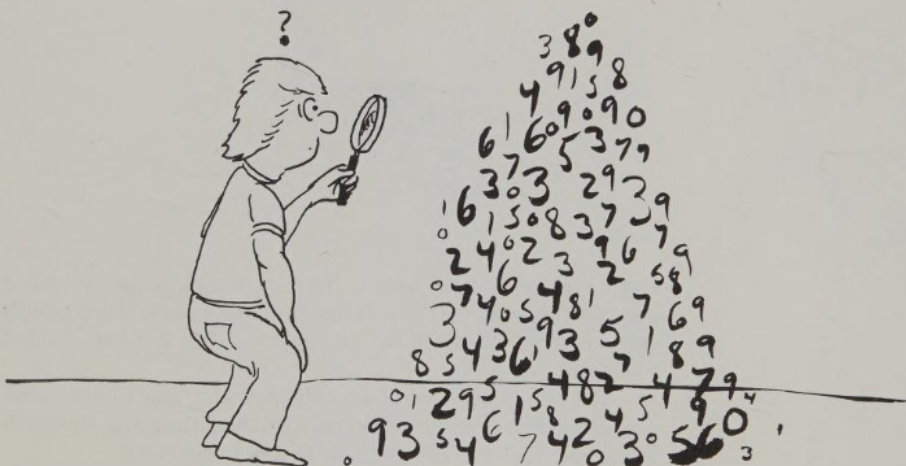
way I ought to go from here?", the cat answers, "That depends a good deal on where you want to get to." Alice says, "I don't much care where,—" to which the cat replies, "Then it doesn't matter which way you go."

The scientists, according to Mr. Kemeny, aren't in the business of deciding which direction mankind should be going. But when we decide, they can tell us the best way to get there.

Here's your chance to play detective with some numbers. The program in this adventure will ask you for a number called alpha. It



will take your alpha and turn it into a number called beta. These numbers are related in some way. That is, there is some mathematical operation that the computer does to alpha to get beta. For instance if your alpha equals 2 and the beta is 7, the operation would be add 5. The lines in the program that set the rule are line 110 and 120. Since you know the rule by typing in the program, this is no good to do by yourself. See if you can challenge your friends to change lines 110 and 120. Then run the program for someone who doesn't know the rule. Good luck and good number-hunting!



Program Listing

```
 10 HOME
 20 PRINT "THIS PROGRAM WILL TEST YOUR"
 30 PRINT "DETECTIVE SKILLS WITH NUMBERS."
 40 PRINT
 50 PRINT "YOU WILL ENTER 'ALPHAS', AND"
 60 PRINT "THE COMPUTER WILL DO SOMETHING"
 70 PRINT "TO THE ALPHA TO TURN IT INTO"
 80 PRINT "A 'BETA'.  YOUR MISSION:"
 90 PRINT "TO FIND OUT WHAT THE COMPUTER"
100 PRINT "HAS DONE TO YOUR ALPHA."
110 m = INT(10*RND(1))+1
120 i = INT(10*RND(1))+1
130 PRINT
140 PRINT "WHAT IS YOUR VALUE FOR ALPHA?"
150 INPUT a
160 b = m*a+i
170 PRINT "IF ALPHA= "; a; " THEN BETA= "; b
180 PRINT
190 PRINT "DO YOU NEED ANOTHER CLUE?"
200 INPUT a$
```



```
210 IF a$ = "y" OR a$ = "Y" OR a$ = "yes" OR a$ = "YES" THEN 140
220 PRINT
230 PRINT "O.K., LET'S SEE HOW YOU DID..."
240 PRINT "YOUR ALPHA WAS MULTIPLIED"
250 PRINT "BY "; m; " AND THEN "; i; " WAS ADDED TO"
260 PRINT "IT TO GET THEN VALUE FOR BETA."
270 PRINT
280 PRINT "DO YOU WANT TO TRY ANOTHER MISSION?"
290 INPUT a$
300 IF a$ = "y" OR a$ = "Y" OR a$ = "yes" OR a$ = "YES" THEN 50
310 END
```

Explanation

The variable list is shown here. The line-by-line description follows it.

m	the multiplier of alpha
i	the addition to alpha
a	alpha (your value)
b	beta (the computer's value)
a$	yes/no answers

Lines 10-100. Prints the introduction and explanation.

Lines 110-120. Chose random numbers from 1 to 10 to use as m and b. (Here's where you can change the program to make it harder. Change the 10s to 100s or 1000s, and it will use bigger numbers.)

Lines 130-150. Asks for your value for alpha.

Lines 160-180. Calculates beta from your alpha and m and b, and displays it for you.

Lines 190-210. Asks if you need another alpha-beta pair.

Lines 220-270. If you need no more pairs, shows you the numbers it used, then asks if you'd like another mission.

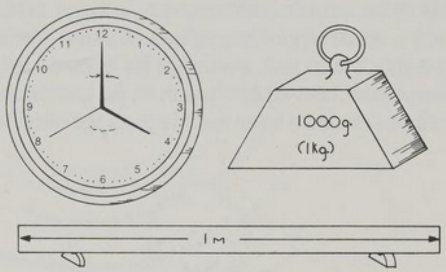
Lines 280-300. Asks if you'd like another mission, and goes back to line 50 to pick new random numbers if you do.

Sample Run

```
 THIS PROGRAM WILL TEST YOUR
DETECTIVE SKILLS WITH NUMBERS.

YOU WILL ENTER 'ALPHAS', AND
THE COMPUTER WILL DO SOMETHING
TO THE ALPHA TO TURN IT INTO
A 'BETA'.  YOUR MISSION:
TO FIND OUT WHAT THE COMPUTER
HAS DONE TO YOUR ALPHA.
```

```
WHAT IS YOUR VALUE FOR ALPHA?
?0
 IF ALPHA= 0 THEN BETA= 5

DO YOU NEED ANOTHER CLUE?
?y
WHAT IS YOUR VALUE FOR ALPHA?
?1
IF ALPHA= 1 THEN BETA= 13

DO YOU NEED ANOTHER CLUE?
?y
WHAT IS YOUR VALUE FOR ALPHA?
?2
IF ALPHA= 2 THEN BETA= 21

DO YOU NEED ANOTHER CLUE?
?n

O.K., LET'S SEE HOW YOU DID...
YOUR ALPHA WAS MULTIPLIED
BY 8 AND THEN 5 WAS ADDED TO
IT TO GET THE VALUE FOR BETA.

DO YOU WANT TO TRY ANOTHER MISSION?
?n
```


6



Adventures in Physics

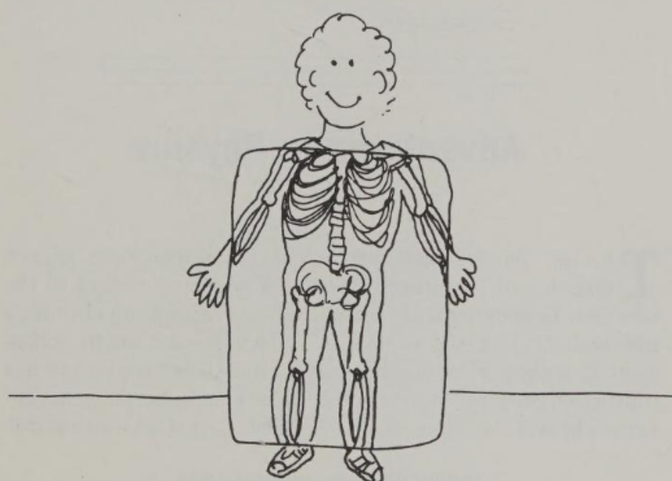
This last (but not least) chapter will take us deeper into science than any of the others. Physics is at the root of all of the sciences. Despite what is obvious, biology depends on chemistry and chemistry depends on physics. You will see some of the fundamentals of physical motion and energy, and we've even thrown in a conversion program to help you both here and elsewhere in converting between systems of measurement. Last stops—all aboard!

FREQUENCY AND WAVELENGTH

All energy in the universe is constantly trying to get somewhere. Some types move quickly, some move slowly, and some are temporarily stalled in the form of mass.

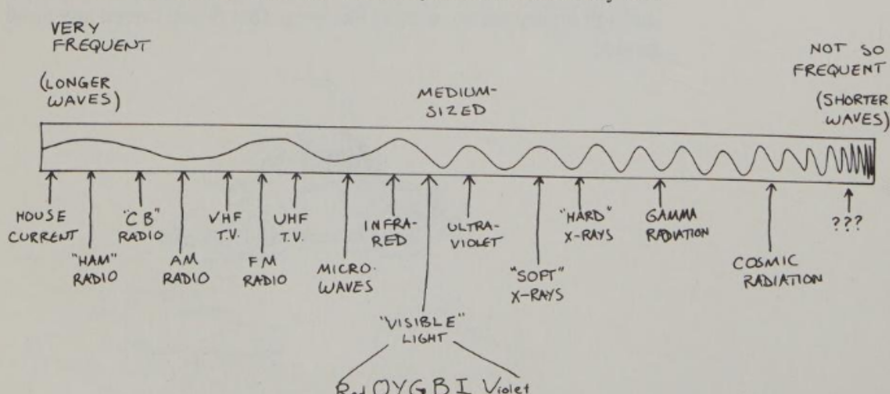
One large group of energies is known as *Electromagnetic Radiation* (EMR). They hold the distinction of being able to travel through empty space, sort of like a car that doesn't need any solid ground.

Needless to say, they are quite strange. The behavior of this type of radiation has puzzled physicists for decades. It turns out that they are very useful. Some of the more popular EMRs include radio waves, television signals, microwaves, infrared, visible light, ultraviolet rays, x-rays, gamma rays, and some cosmic radiation. You've probably come into contact with or used most of these. Infrared keeps your hamburgers warm at the restaurant, and x-rays allow doctors to examine your insides from the outside. Utraviolet

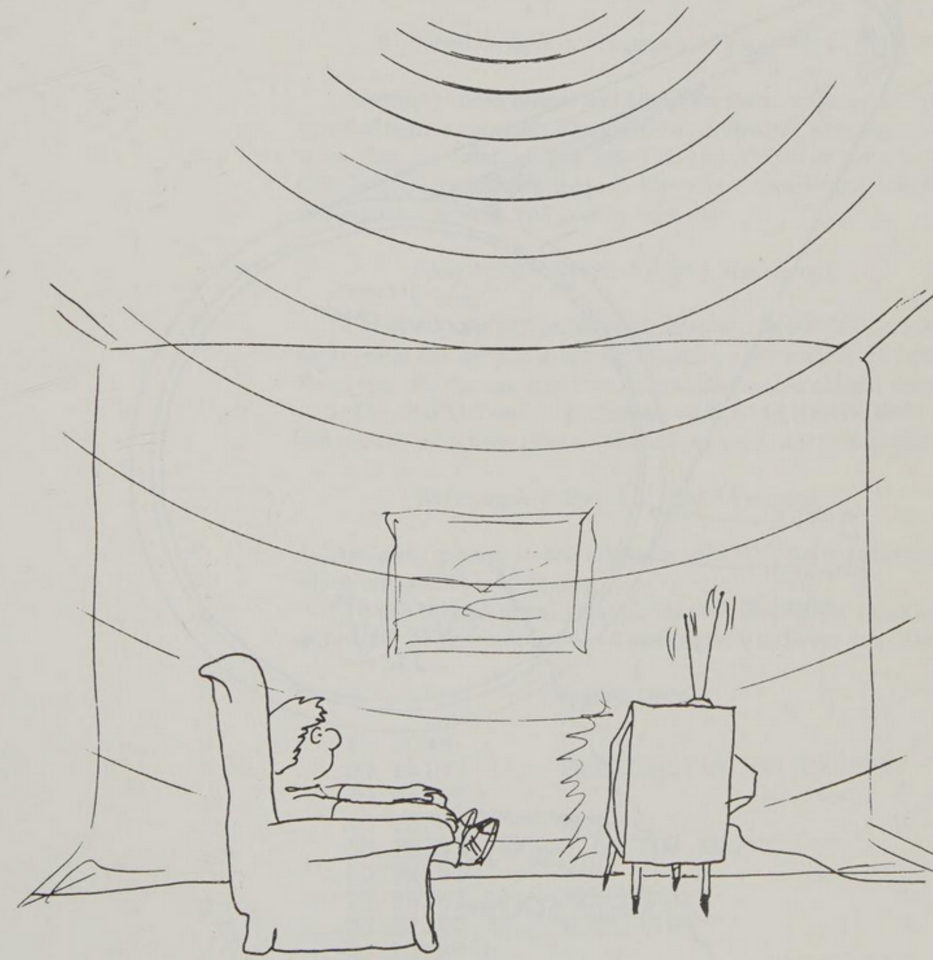


helps to sterilize hospital equipment, and gamma radiation is a by-product of nuclear power generation.

All of these forms are closely related. In fact, they are so similar, that they have been known to trade places, and convert from one to the other. This chart shows just how close they are:

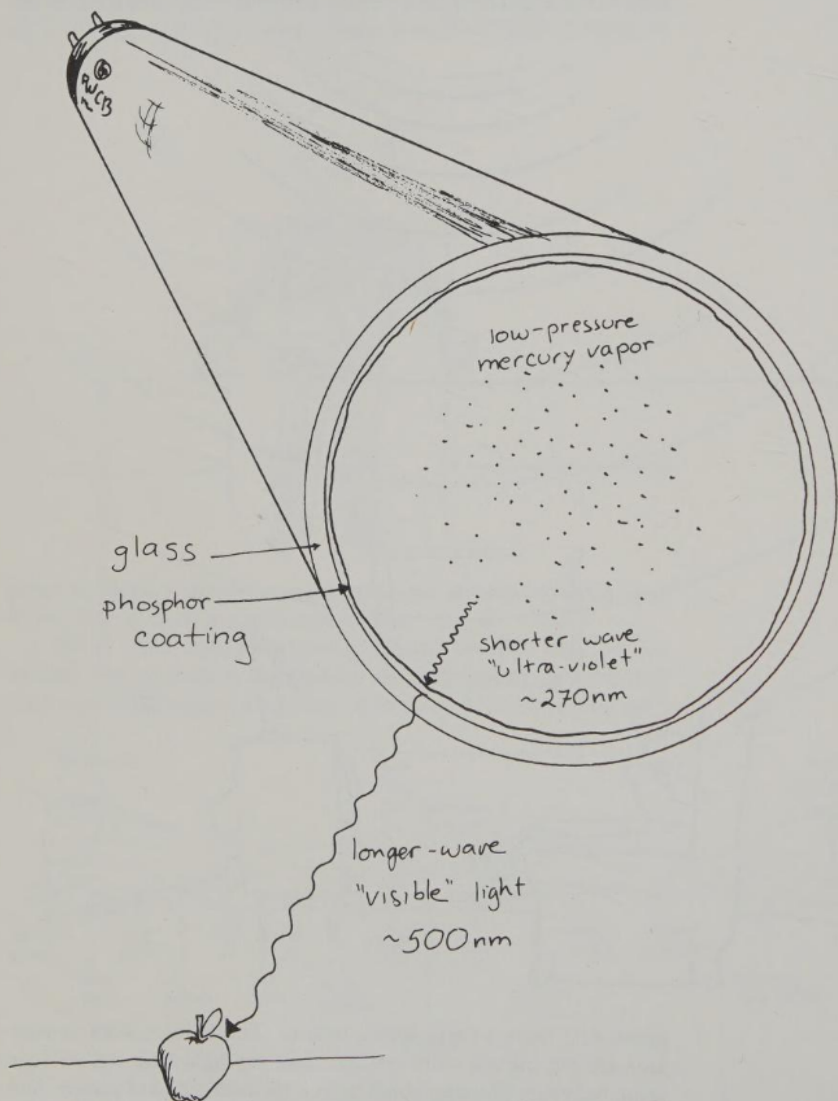


What we call visible light is smack dab in the middle of this spectrum. The reason that we can see this portion of the spectrum is because we have eye pigments which vibrate in harmony (resonate) with the waves of light and, in turn, stimulate our brain to think red or green. The others EMRs are invisible to us, being either too long or too short to resonate our pigments. Because they aren't visible doesn't mean that they aren't there. Turn on a radio or tv and you can



prove that there's radio waves nearby. The radio towers in your area are putting out more wattage than any light bulb you've ever seen, but your eyes aren't built to pick up those types of waves. And while x-rays go through you and register on film, that machine in the dentist's office never seems to do much more then click and buzz.

The main difference between these waves is in their length. Like we said, the length can change. The radiation in a fluorescent bulb starts out as ultraviolet, too short for us to see. But when it hits the phosphor coating inside the bulb, it is re-released as longer waves that are visible to us. Plants do something similar. They soak



up sunlight, which is visible. Any extra sunlight is acted upon by the green pigments so that it is re-released (fluoresced) as far-red, too long for us to see. So green plants actually glow red when they're in the sunlight.

How long or short are the waves which we can see? Our eyes can't see anything shorter than 300 billionths of a meter or longer than 700 billionths of a meter. What does that equal in vibrations? To find out, you need this equation:

Speed of light = Frequency × Wavelength

Thanks to Oleas Romer and Albert Einstein, we know that the speed of light is constant. Thanks to James Bradley and others, we know that the value of this speed is 186,000 miles per second (299,790,000 meters per second). If you have wavelength, just flip this equation around, and you can see that:

Frequency = Speed of light / Wavelength

Our eye's .0000003 meter limit turns into 99,990,000,000,000 hertz (we'll use *hertz* to stand for vibrations per second. Heinrich Hertz was the person who discovered this motion of light waves . . .). The .0000007 meter limit equals 42,827,143,000,000 hertz. If you need to know wavelength, use the equation in this arrangement:

Wavelength = Speed of light / Frequency

An AM radio station at 680 kilohertz (680,000) hertz has waves which are 440.86 meters long.

To use this program, give ADAM the name of the radiation, and either the frequency or the wavelength, whichever you have.

Program Listing

```
 10 HOME
 20 PRINT "    FREQUENCY & WAVELENGTH"
 30 PRINT
 40 PRINT
 50 PRINT "WHICH DO YOU KNOW?"
 60 PRINT
 70 PRINT "1.  FREQUENCY"
 80 PRINT "2.  WAVELENGTH"
 90 INPUT a
100 ON a GOTO 110, 260
110 PRINT
120 PRINT "NAME THE RADIATION."
130 PRINT "TYPE (RED LIGHT, ULTRAVIOLET,"
140 PRINT "TV, X-RAY, ETC.)"
```

```
150 INPUT r$
160 PRINT
170 PRINT "WHAT IS "; r$; "'S"
180 PRINT "FREQUENCY IN CYCLES PER SEC"
190 PRINT "(CALLED HERTZ)?"
200 INPUT f
210 l = 299000000/f
220 PRINT
230 PRINT "A SINGLE WAVE OF "; r$; " AT "
240 PRINT l; " METERS LONG."
250 GOTO 370
260 PRINT "NAME THE RADIATION."
270 PRINT "TYPE (RED LIGHT,ULTRAVIOLET,"
280 PRINT "TV, X-RAY, ETC.)."
290 INPUT r$
300 PRINT
310 PRINT "WHAT IS "; r$; "'S"
320 PRINT "WAVELENGTH IN METERS?"
330 INPUT l
340 f = 299000000/l
350 PRINT
360 PRINT "THE FREQUENCY OF A "; l; " METER"
370 PRINT r$; " WAVE IS "; f
380 PRINT "CYCLES PER SEC."
390 END
```

Explanation

The variables list and the explanation of the program are given here.

a	choice of conversions
r$	radiation's name
l	wavelength of r$
f	frequency of r$

Lines 10-40. Prints the title.

Lines 50-100. Asks if you know the frequency or wavelength, and sends you to the proper subroutine. If you answer 1, it goes to 110, if you answer 2 it goes to 260.

Lines 110-150. Asks for the radiation's type (its name).

Lines 160-200. Asks for the frequency of the radiation in hertz (cycles per second).

Line 210. Calculates wavelength according to $c = l*f$ (where $c = 2.99 \times 10^8$ centimeters per second).

Lines 220-240. Displays the result.

Line 250. Goes to the end of the program.

Lines 260-290. Asks for the type (name) of the radiation.

Lines 300-330. Asks for the radiation's wavelength (in meters: remember, one centimeter is .01 meter, one millimeter is .001 meter, etc.).

Line 340. Calculates frequency based on $c = l*f$.

Lines 350-380. Displays the result.

Line 390. The end.

Sample Run

```
        FREQUENCY & WAVELENGTH


WHICH DO YOU KNOW?

1.  FREQUENCY
2.  WAVELENGTH
?2
NAME THE RADIATION.
TYPE (RED LIGHT,ULTRAVIOLET,
TV, X-RAY, ETC.).
?ULTRAVIOLET

WHAT IS ULTRAVIOLET'S
WAVELENGTH IN METERS?
?3E-03

THE FREQUENCY OF A 3E-03 METER
ULTRAVIOLET WAVE IS 9.96666666E+10
CYCLES PER SEC.
```

SPEED OF LIGHT

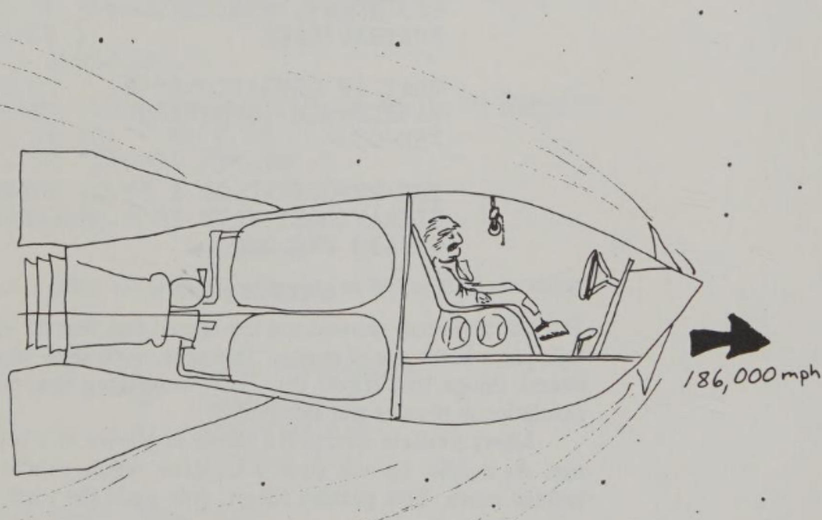
The last adventure showed you the crucial part that the speed of light plays in the use of energy. This time, we'll show you how it affects things that travel very fast—something that caused a revolution in physics and astronomy.

Albert Einstein started his career in science at a very early age. As a child, he was given a compass, whose needle always pointed north. This puzzled Albert: how could the earth's poles

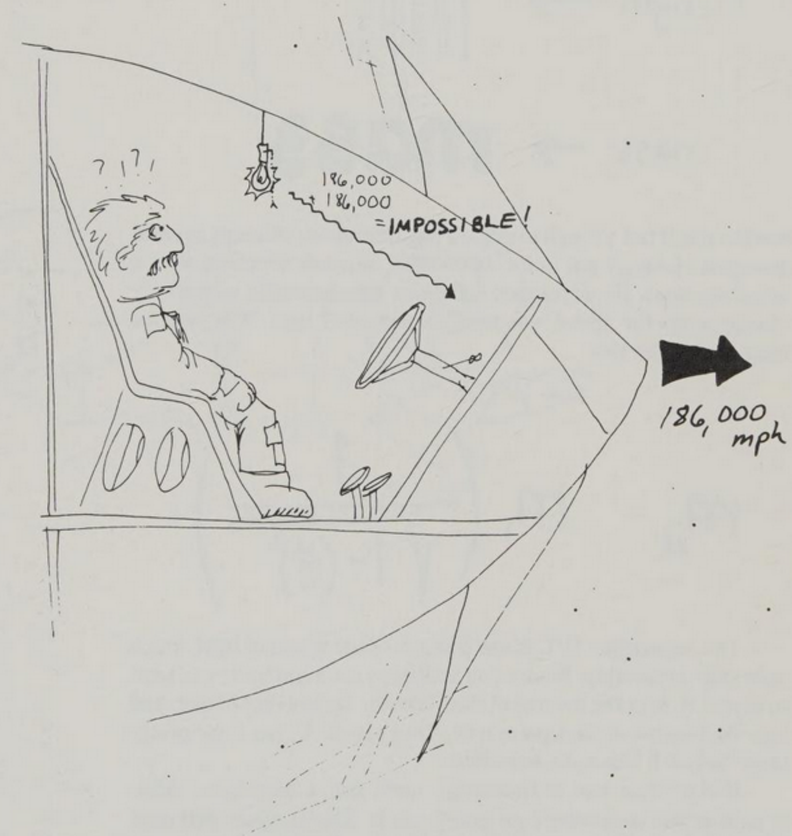
cause the compass to move without touching it?

This is what Albert set out to solve. He never figured it out. To this day, nobody has. In that sense, Einstein was a failure. But while digging for the answer, he uncovered secrets of nature more precious than any found before.

Einstein was, by his own admission, a miserable mathematician. He did have the courage to seek out excellent teachers to train him as needed. He also compensated for his math skills with an extraordinary imagination, and a remarkable ability to think through a problem that was difficult to describe in numbers. One of his most challenging thought experiments concerned riding in a spaceship.

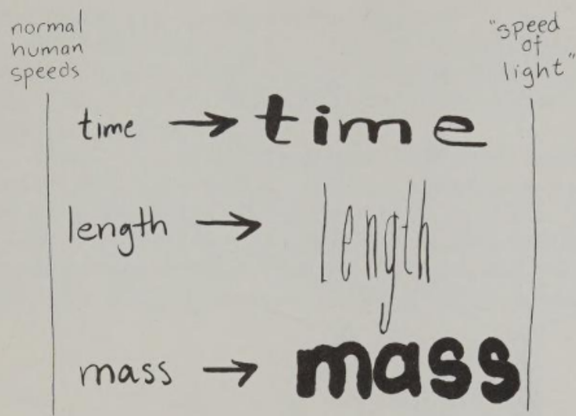


Albert knew that the speed of light was the fastest anything could go. If his imaginary rocket sped up until it reached the speed of light, several things would happen. One was that lights on board would stop working (their waves could never go fast enough to hit things at the front of the ship). This was against one of the most important laws of physics. *Relativity* said that you can't tell what your sur-



roundings are doing unless you go outside of them. The only way that a spaceship could travel near the speed of light is if the ship itself changed length as it sped up, allowing the light inside to keep acting normally. This change is known as the *Lorentz Transformation*. There's another transformation for time, called *dilation*, and a transformation for mass. That's the one that we're going to look at.

Einstein knew that length, time, and mass would all change when an object moved fast enough. The speeds that we humans are

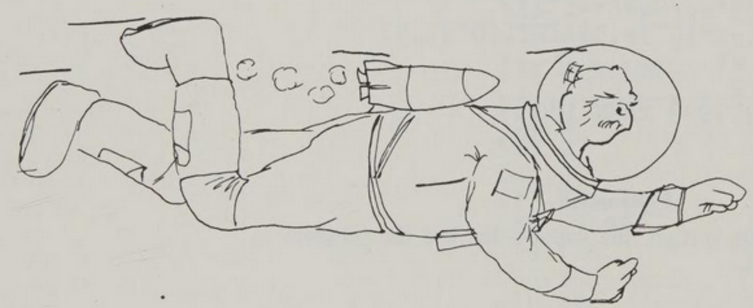


used to aren't fast enough to make a big difference (although several astronauts jokingly put in for .000000001 seconds overtime after a moon mission). He developed equations whose results would only change when the speed was near the speed of light. The one for mass looks like this:

$$m_2 = m_1 \left(\frac{1}{\sqrt{1-\left(\frac{u}{c}\right)^2}} \right)$$

The expression U/C is the fraction of the speed of light which your ship is traveling. Before you multiply your mass by it, you have to square it, take the inverse of that, take the square root of that, and take the inverse of the square root. Your speed, U, has to be pretty large before it can have any effect.

Rather than use a spaceship, we'll use a simulator. After keying-in this adventure's program, run it. The program will start by asking you for your weight, and will calculate your mass in kilograms. You will then see what happens to this mass as you accelerate towards the speed of light. Each time you accelerate, you will increase by one-half of what's left between your speed and the speed of light. You'll never actually reach the speed of light by this method (Why not?). That's alright, though, because one of the other "gems" of physics that Einstein uncovered was that nothing that is an object can go as fast as the speed of light. Not convinced? The speed controller is in line 190. Try to change it and put in your own



values above the speed of light. What happens?

Who knows where your adventures in physics may lead? One of the greatest journeys was started with nothing more than a compass.

Program Listing

```
10 HOME
20 PRINT " MASS CHANGE NEAR LIGHT SPEED"
30 PRINT
40 PRINT
50 PRINT "WHAT IS YOUR WEIGHT IN POUNDS?"
60 INPUT w
70 mo = w/2.205
80 PRINT
90 PRINT "    MPH", "  WEIGHT"
100 PRINT "(IN MILLIONS)", "(IN POUNDS)"
110 PRINT "-----------------------------"
120 f = .5
130 s = f*.186*60*60
140 s = INT(s*10^3+.5)/INT(10^3+.5)
150 wt = 2.205*mo/SQR(1-f^2)
160 wt = INT(wt*10^3+.5)/INT(10^3+.5)
170 PRINT s, wt
180 f = .5+f/2
190 IF f < .999999 THEN 130
200 END
```

Explanation

In the following section, the variable list and the program description is given.

w	your weight in pounds
mo	your mass in kilograms
f	fraction of speed of light
s	speed in miles per hour
wt	weight at given speed

Lines 10-40. Prints the introduction.

Lines 50-60. Asks your weight in pounds.

Line 70. Converts weight in pounds to mass in kilograms (2.205 pounds/kilogram).

Lines 80-110. Prints the header for speed/mass chart.

Line 120. Sets fraction of speed of light to ½ for first example.

Line 130. Converts this fraction to miles per hour (times 186,000 divided by 1 million, times seconds per minute, times minutes per hour.

Line 140. Rounds speed to three decimal places.

Line 150. Calculates weight at that speed 2.205*mo converts to pounds; /SQR($1-f^2$) raises f to the second power, subtracts that from 1, takes its square root, then takes its inverse (also known as the Lorentz Transformation for Mass).

Line 160. Rounds to three decimal places.

Line 170. Prints your speed and weight.

Line 180. Raises f by one-half of what's "left" between it and 1.

Lines 190-200. Checks to see if f is too close to 1 (and no more change in weight takes place). If it is, the program ends. If not, it continues at line 130.

Sample Run

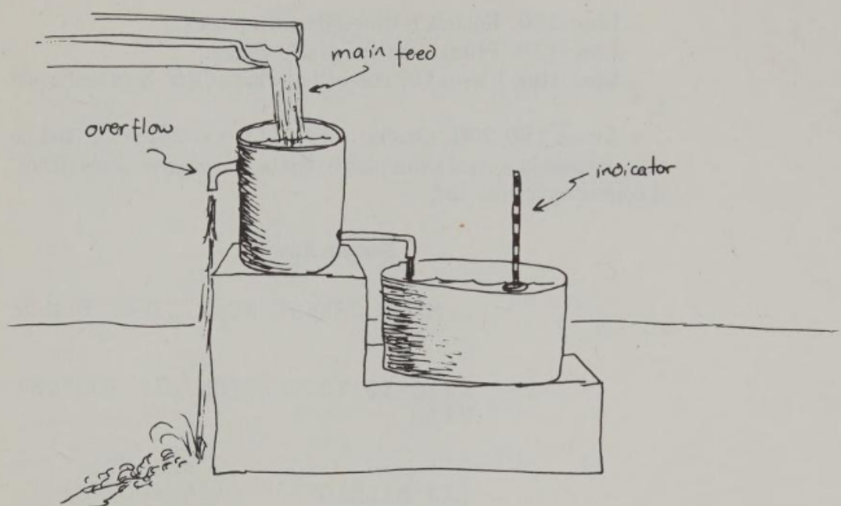
```
     MASS CHANGE NEAR LIGHT SPEED


WHAT IS YOUR WEIGHT IN POUNDS?
?110

    MPH              WEIGHT
(IN MILLIONS)    (IN POUNDS)
------------------------------
334.8            127.017
502.2            166.305
585.9            227.215
627.75           316.104999
648.674999       443.478
659.138          624.699
664.369          881.724
666.984          1245.725
668.292          1760.86
668.945999       2489.624
669.273          3520.43
669.437          4978.336
669.517999       7040.215
669.559          9956.215
669.58           14080.107
669.59           19912.203
669.595          28160
669.596999       39824.254
669.599          56320
```

PENDULUMS

Clocks are not easy to engineer. The key to a good clock is to find something that keeps doing the same thing reliably and with very little change in its rate.

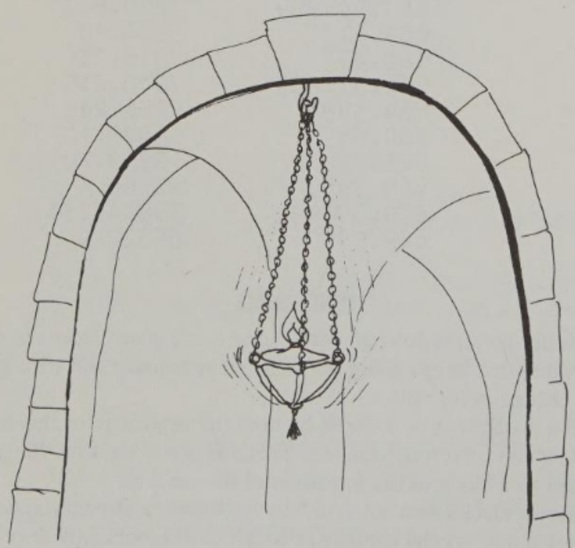
No doubt, you've noticed by now the regularity of the sun's apparent motion through the sky. This was one of the first things to be used as a clock in the invention of the sundial.

Water clocks were soon to follow. Rivers or streams could be channeled into special containers which would constantly drip into



an indicating chamber. As long as there were no heavy rains or long droughts, these *clepsydras* kept good time. They did, however, require plumbing, and could not be transported or used in extreme cold.

One of the first transportable clocks was the pendulum clock. The virtues of the pendulum as a timekeeper were first reported by Galileo Galilei. He noticed a lamp suspended from the ceiling of a

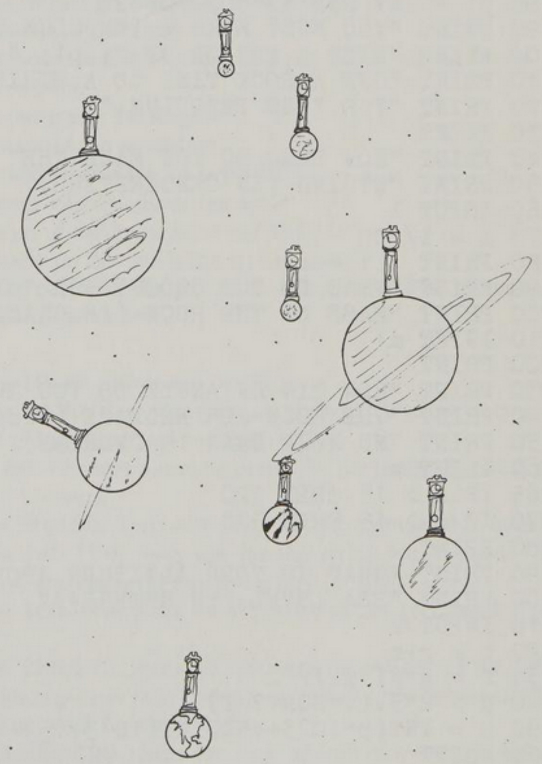


church. The swinging of the lamp was very regular as Galileo timed it with his pulse rate. Galileo then built models of pendulums to find out more about them. He found that four things dictated how fast the pendulum would swing:

- ☐ The length of the string.
- ☐ The mass of the pendulum.
- ☐ The angle of the swing.
- ☐ The local gravity.

If you change any of these things, the period it takes the pendulum to swing will change. And if you choose these things carefully, you can make a pendulum of any period.

This program will tell you the period resulting from any pendulum with the length, mass, angle, and gravity that you specify. With the numbers in the weights on different planets adventure, you



can write your own new adventure, Pendulums On Different Planets. With a little back-solving, you can chose a period and find out what length, mass, and angle you need. You may want to build a clock of your own. Pendulum clocks are fairly accurate and are still quite common. The grandfather and regulator wall clocks you may have seen are pendulum clocks and were the mainstay of this country's railroads for many years.

Program Listing

```
 10 HOME
 20 PRINT "           PENDULUMS"
 30 PRINT
 40 PRINT
 50 m2 = 5.98E+24
 60 g = 6.67E-11
 70 r = 6378000
 80 p1 = INT(RND(1)*30)/10+.2
 90 PRINT "YOU MUST MAKE A PENDULUM SWING"
100 PRINT "WITH A PERIOD OF "; p1; " SECONDS."
110 PRINT "USE A ROCK TIED TO A STRING"
120 PRINT "FOR YOUR PENDULUM."
130 PRINT
140 PRINT "HOW LONG DO YOU MAKE THE"
150 PRINT "STRING (IN CENTIMETERS)?"
160 INPUT l
170 l = l/100
180 PRINT
190 PRINT "WHAT DO YOU CHOOSE FOR THE"
200 PRINT "MASS OF THE ROCK (IN GRAMS)?"
210 INPUT m1
220 PRINT
230 PRINT "HOW BIG AN ANGLE DO YOU MOVE"
240 PRINT "THE ROCK FOR RELEASE (MUST BE"
250 PRINT "NO MORE THAN 15 DEGREES)."
260 INPUT a
265 IF a > 15 THEN 220
270 IF a > 15 THEN 220
280 PRINT
290 PRINT "WHAT IS YOUR ALTITUDE ABOVE"
300 PRINT "SEA LEVEL (IN METERS)?"
310 INPUT z
320 r = r+z
330 f = g*m1*m2/r^2
340 p = 2*3.14*SQR(l/f)
350 p = INT(p*10^3+.5)/INT(10^3+.5)
360 PRINT
370 PRINT "------------------------------"
380 PRINT "             RESULTS"
390 PRINT "------------------------------"
```



```
400 PRINT "STRING LENGTH:  "; l*100
410 PRINT "MASS:  "; m1
420 PRINT "ANGLE:  "; a
430 PRINT "ALTITUDE:  "; z
440 PRINT "PERIOD:  "; p
450 IF ABS(p-p1)/p1 < .02 THEN 500
460 PRINT
470 PRINT "NOT CLOSE ENOUGH!  TRY AGAIN!"
480 PRINT
490 GOTO 130
500 PRINT "CLOSE ENOUGH!  GOOD FOR YOU!"
510 END
```

Explanation

The variable list and line-by-line description are shown here.

m2	the mass of the earth
g	universal gravitational constant
r	radius of the earth
p1	random period of swing of a pendulum, picked by the computer
l	your guess for length
m1	mass of the pendulum
a	angle of release for pendulum
z	your altitude above sea level
r	sum of earth diameter and altitude
f	force on the pendulum due to gravity
p	the period of the pendulum that your length, altitude, and mass make

Lines 10-40. Prints the heading.

Lines 50-70. Set values for the mass of the earth, radius of the earth, and gravitational constant.

Line 80. Chooses a random period for the pendulum between 0.2 and 3.2 seconds.

Lines 90-130. Tells you the period that you need to match.

Lines 140-170. Asks you for the string length needed in centimeters, and converts this to meters (l/100).

Lines 180-210. Asks for the mass of the rock needed (in grams).

Lines 220-270. Asks for the angle of release of the pendulum. If this is greater than 15 degrees, it asks again because these formulas only work for small angles.

Lines 280-320. Asks for your altitude-above-sea-level in meters (if you only know feet, there are 3.28 feet per meter), and adds this to the earth radius (the equations need to know how far you are from the earth's center).

Line 330. Calculates the force due to gravity on your pendulum based on the two masses, the gravitational constant, and the radius of the earth (just like in Escape Velocity and Weights On Different Planets).

Lines 340-350. Calculates and rounds-off the period of the pendulum which you chose based on the length and force calculated in lines 330 and 160. 2 * 3.14 is twice the value for pi, the ratio between a circle's perimeter and diameter. This is used because of the cycle-like motion of the pendulum, and two-pi is the amount of motion you go through in one full cycle of rotation or vibration.

Lines 360-440. Prints your guess and the results.

Line 450. Checks to see if the fraction of error is close enough (less than 1 in 1000). If it is, it goes to line 500. If not, it goes to line 460.

Lines 460-490. Prints the message for large errors, and then re-asks for guesses, starting at line 130.

Lines 500-510. Prints message, and then ends the program.

Sample Run

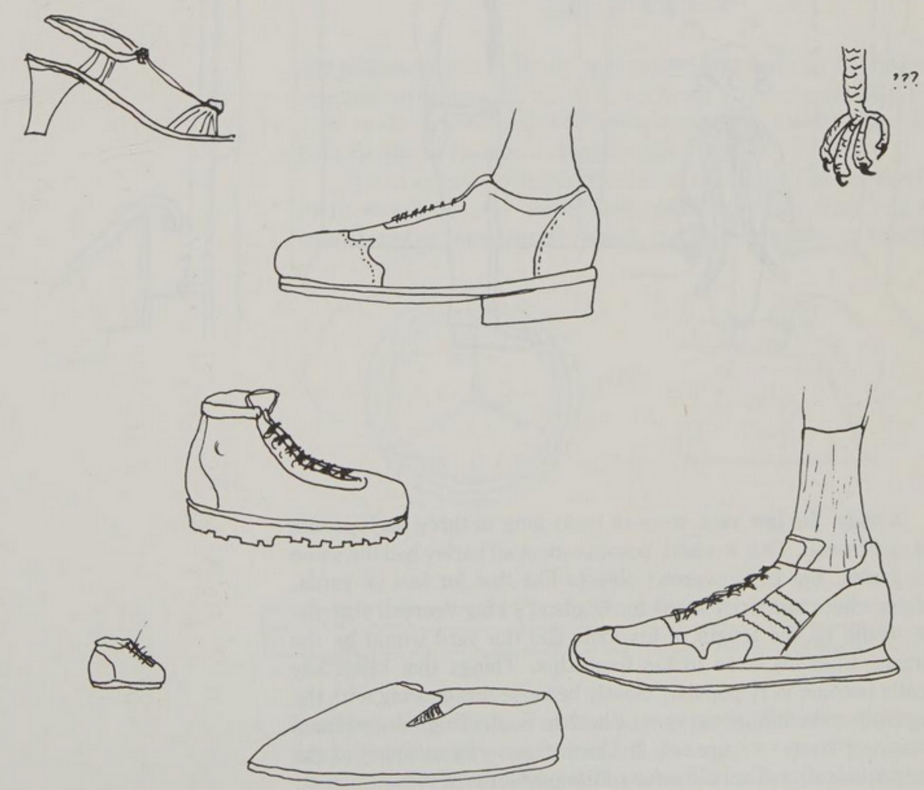
```
           PENDULUMS


YOU MUST MAKE A PENDULUM SWING
WITH A PERIOD OF 2.3 SECONDS.
USE A ROCK TIED TO A STRING
FOR YOUR PENDULUM.

HOW LONG DO YOU MAKE THE
STRING (IN CENTIMETERS)?
?25

WHAT DO YOU CHOOSE FOR THE
MASS OF THE ROCK (IN GRAMS)?
?50

HOW BIG AN ANGLE DO YOU MOVE
THE ROCK FOR RELEASE (MUST BE
NO MORE THAN 15 DEGREES).
?10

WHAT IS YOUR ALTITUDE ABOVE
SEA LEVEL (IN METERS)?
?100

------------------------------
           RESULTS
------------------------------
```



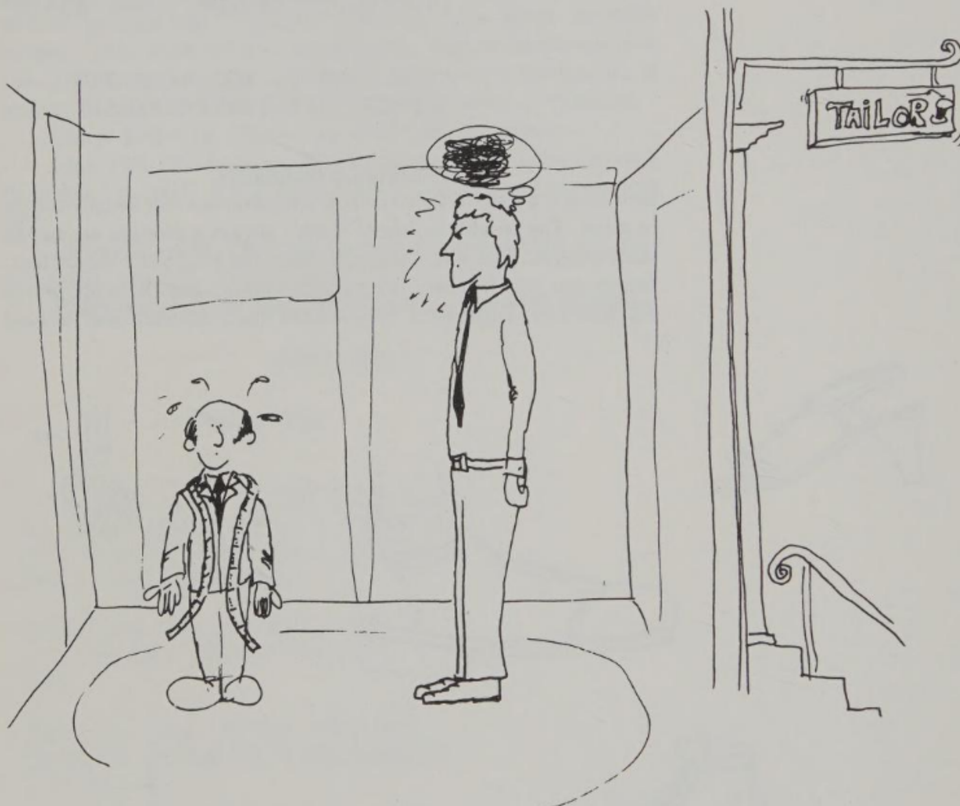
```
STRING LENGTH:  25
MASS:  50
ANGLE:  10
ALTITUDE:  100
PERIOD:  .142

NOT CLOSE ENOUGH!  TRY AGAIN!


HOW LONG DO YOU MAKE THE
STRING (IN CENTIMETERS)?
?
```

METRIC CONVERSIONS

How long is a foot? Twelve inches is the number that usually comes to mind. The trouble is, there wasn't always a straight answer to that question. The unit foot is just that—the length of a human foot. But anyone who has ever bought shoes knows that there are lots of different feet. Yards were measured in much the same way. A yard

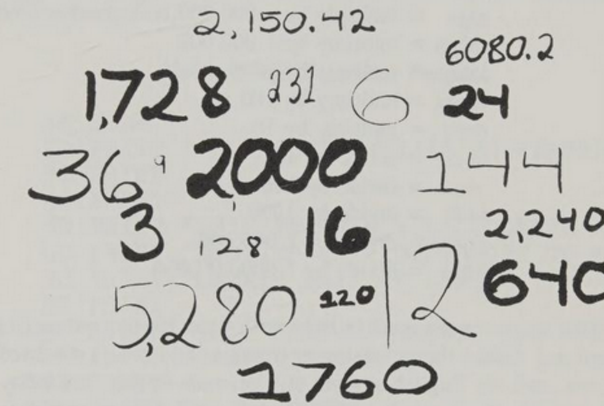


was the distance from your nose to your outstretched fingers. This was handy (no pun intended) when measuring cloth or rope, as you pulled it off of a bolt or out of a coil. But what if the seller was half as tall as the buyer? Or if your real estate agent had small feet? Something had to be done about it. A standard had to be used.



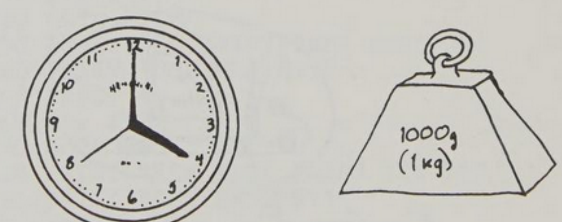
Inches, the law said, were to be as long as three barleycorns laid end-to-end. That worked, because most all barley had the same size grains. But there weren't objects like that for feet or yards. This is where authority steps in. England's king decreed that the foot would be the length of his foot, and the yard would be the distance from his nose to his fingertips. Things that kings say usually become very popular, mostly because disagreeing with the king tends to be hazardous to your health. So the English system of measuring started to spread. It became adopted in much of the western world, and solved most of the problems of measurement.

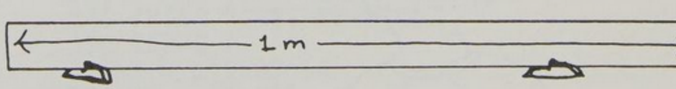
The system itself had one big problem, though. Most of the units were invented independently, with all sorts of conversions between units. The factor between ounces to pounds was 16. Between pounds and tons, the conversion was 2000. Between pints



and gallons, it was 8. Between feet and inches, it was 12. Five and one-half yards equaled a rod, and 40 rods equaled a furlong. Four rods made 1 acre, and 640 acres made one square mile. This hodgepodge of numbers can get maddening.

A new system of measurement was set up. It was decided that there would be only three basic units; seconds (time), grams (mass), and meters (length). Water, the universal solvent, would be





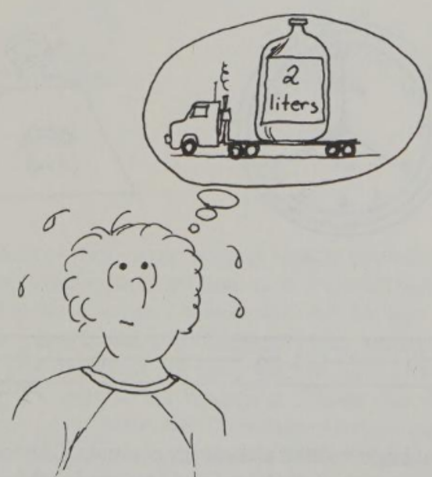
used to calibrate volume and energy contents. The meter (from the Latin, to measure) would be defined first as a fraction of the distance between the north and south poles, later as the length of a platinum-iridium bar, and eventually as a multiple of the wavelength of a vibrating cesium atom (sure beats barleycorns).

Best of all, there would be only one conversion factor: 10! Prefixes would be used to make larger or smaller units:

giga = multiply by 1,000,000,000
mega = multiply by 1,000,000
kilo = multiply by 1000
centa = multiply by 100
deca = multiply by 10
deci = divide by 10
centi = divide by 100
milli = divide by 1000
micro = divide by 1,000,000
nano = divide by 1,000,000,000

This system goes hand-in-hand with base 10, our numbering system and makes things easier to convert. Old habits die hard, however, and the English system still prevails in America today. Oddly enough, the English don't use the English system anymore. They use the metric system. The metric system does get a lot of use in science, because it fits right in with scientific notation.

Most of us, however are used to the English system. Even if you have learned the metric system, you may not have a good feel for the physical reality of the quantities. We all need to be able to get a mental picture of 6 kilometers, or a kilogram of sugar, or 2 liters of



soda. Getting those pictures in your mind comes only with experience. But there will be times when you need to produce figures in metrics (like in a lot of these adventures) but all of your information is in English units. That's what this program is for. ADAM will help you to convert between systems and allow you to use equations in both systems.

Metric Length Converter Program

```
 10 HOME
 20 PRINT "    ENGLISH/METRIC CONVERTER"
 30 PRINT
 40 PRINT
 50 PRINT "THIS PROGRAM WILL CONVERT AN "
 60 PRINT "ENGLISH UNIT OF LENGTH TO A"
 70 PRINT "METRIC UNIT OF LENGTH."
 80 PRINT
 90 PRINT "WHAT UNIT OF LENGTH ARE YOU"
100 PRINT "CONVERTING FROM?"
110 PRINT
120 PRINT "1.  INCHES"
130 PRINT "2.  FEET"
140 PRINT "3.  YARDS"
150 PRINT "4.  MILES"
160 INPUT ul
170 ON ul GOTO 180, 230, 280, 330
180 PRINT
190 PRINT "HOW MANY INCHES ARE YOU"
200 PRINT "CONVERTING?"
210 INPUT i
220 GOTO 370
230 PRINT "HOW MANY FEET ARE YOU"
240 PRINT "CONVERTING?"
250 INPUT f
260 i = f*12
270 GOTO 370
280 PRINT "HOW MANY YARDS ARE YOU"
290 PRINT "CONVERTING?"
300 INPUT y
310 i = y*3*12
320 GOTO 370
330 PRINT "HOW MANY MILES ARE YOU"
340 PRINT "CONVERTING?"
350 INPUT m
360 i = m*5280*12
370 mm = 25.4*i
380 cm = 2.54*i
390 dm = .254*i
400 m = .0254*i
410 km = 2.54E-05*i
```

```
420 PRINT
430 PRINT "THAT IS:"
440 PRINT mm; " MILLIMETERS"
450 PRINT cm; " CENTIMETERS"
460 PRINT dm; " DECIMETERS"
470 PRINT m; " METERS"
480 PRINT km; " KILOMETERS"
490 END
```

Explanation

The explanation of the metric length converter program follows:

ul	units being converted from
i	number of inches to convert
f	number of feet to convert
y	number of yards to convert
m	number of miles to convert
mm	converted to millimeters
cm	converted to centimeters
dm	converted to decimeters
m	converted to meters
km	converted to kilometers

Lines 10-80. Prints the introduction.

Lines 90-150. Displays the menu of conversions.

Lines 160-170. Asks for the units, sends the program to the correct subroutines.

Lines 180-220. Asks for the number of inches which you are converting, and then goes to the conversion routine (since your measure is already in inches, jumps in at line 370).

Lines 230-270. Asks for the number of feet which you are converting, converts to inches, and goes to the conversion routine (lines 360-490).

Lines 280-320. Asks for the number of yards which you are converting, converts to inches, and goes to the conversion routine (lines 360-490).

Lines 330-360. Asks for the number of miles which you are converting, converts to inches, and goes on to the conversion routine.

Lines 370-410. Converts the inches in your measure to millimeters, centimeters, meters, and kilometers. (Notice how they only differ by factors of 10.)

Lines 420-490. Displays the converted measure, and ends.

Metric Volume Converter Program

```
10 HOME
20 PRINT "   ENGLISH/METRIC CONVERTER"
```



```
30 PRINT
40 PRINT
50 PRINT "THIS PROGRAM WILL CONVERT AN "
60 PRINT "ENGLISH UNIT OF VOLUME TO A"
70 PRINT "METRIC UNIT OF VOLUME."
80 PRINT
90 PRINT "WHAT UNIT OF VOLUME ARE YOU"
100 PRINT "CONVERTING FROM?"
110 PRINT
120 PRINT "1.  OUNCES"
130 PRINT "2.  PINTS"
140 PRINT "3.  QUARTS"
150 PRINT "4.  GALLONS"
160 INPUT uv
170 ON uv GOTO 180, 230, 280, 330
180 PRINT
190 PRINT "HOW MANY OUNCES ARE YOU"
200 PRINT "CONVERTING?"
210 INPUT oz
220 GOTO 370
230 PRINT "HOW MANY PINTS ARE YOU"
240 PRINT "CONVERTING?"
250 INPUT p
260 oz = p*16
270 GOTO 370
280 PRINT "HOW MANY QUARTS ARE YOU"
290 PRINT "CONVERTING?"
300 INPUT q
310 oz = q*32
320 GOTO 370
330 PRINT "HOW MANY GALLONS ARE YOU"
340 PRINT "CONVERTING?"
350 INPUT g
360 oz = g*128
370 ml = oz/.033814
375 ml = INT(ml*10^3+.5)/INT(10^3+.5)
380 ml = INT(ml*10^3+.5)/INT(10^3+.5)
385 dl = INT(dl*10^3+.5)/INT(10^3+.5)
390 dl = oz/3.3814
395 l = INT(l*10^3+.5)/INT(10^3+.5)
400 dl = INT(dl*10^3+.5)/INT(10^3+.5)
410 l = oz/33.814
420 l = INT(l*10^3+.5)/INT(10^3+.5)
430 PRINT
440 PRINT
450 PRINT "THAT IS:"
460 PRINT ml; " MILLILITERS"
470 PRINT dl; " DECILITERS"
480 PRINT l; " LITERS"
490 END
```

Explanation

The explanation of the metric volume converter follows:

uv	units being converted from
oz	number of ounces to convert
p	number of pints to convert
q	number of quarts to convert
g	number of gallons to convert
ml	converted to milliliters
dl	converted to deciliters
l	converted to liters

Lines 10-80. Prints the introduction.

Lines 90-150. Displays the menu of conversions.

Lines 160-170. Asks for the units, and sends the program to the correct subroutines.

Lines 180-220. Asks for the number of ounces which you are converting, and then goes to the conversion routine (since your measure is already in ounces, jumps in at line 370).

Lines 230-270. Asks for the number of pints which you are converting, converts to ounces, and goes to the conversion routine (lines 360-490).

Lines 280-320. Asks for the number of quarts which you are converting, converts to ounces, and goes to the conversion routine (lines 360-490).

Lines 330-360. Asks for the number of gallons which you are converting, converts to ounces, and goes on to the conversion routine.

Lines 370-410. Converts the ounces in your measure to milliliters, deciliters, and liters. (Notice how they only differ by factors of 10.)

Lines 420-490. Displays the converted measure, and ends.

Metric Weight Converter

```
 10 HOME
 20 PRINT "   ENGLISH/METRIC CONVERTER"
 30 PRINT
 40 PRINT
 50 PRINT "THIS PROGRAM WILL CONVERT AN "
 60 PRINT "ENGLISH UNIT OF WEIGHT TO A"
 70 PRINT "METRIC UNIT OF WEIGHT."
 80 PRINT
 90 PRINT "WHAT UNIT OF WEIGHT ARE YOU"
100 PRINT "CONVERTING FROM?"
110 PRINT
120 PRINT "1.  OUNCES"
```



```
130 PRINT "2.  POUNDS"
140 PRINT "3.  TONS"
150 INPUT uw
160 PRINT
170 ON uw GOTO 180, 220, 270
180 PRINT "HOW MANY OUNCES ARE YOU"
190 PRINT "CONVERTING?"
200 INPUT oz
210 GOTO 310
220 PRINT "HOW MANY POUNDS ARE YOU"
230 PRINT "CONVERTING?"
240 INPUT p
250 oz = p*16
260 GOTO 310
270 PRINT "HOW MANY TONS ARE YOU"
280 PRINT "CONVERTING?"
290 INPUT t
300 oz = t*32000
310 g = oz*28.35
320 kg = oz*.02835
330 kg = INT(kg*10^3+.5)/INT(10^3+.5)
340 mt = oz*3.176E-05
350 mt = INT(mt*10^3+.5)/INT(10^3+.5)
360 PRINT
370 PRINT
380 PRINT "THAT IS:"
390 PRINT g; " GRAMS"
400 PRINT kg; " KILOGRAMS"
410 PRINT mt; " METRIC TONS"
420 END
```

Explanation

The explanation of the metric weight converter follows:

uw	units being converted from
oz	number of ounces to convert
p	number of pounds to convert
t	number of tons to convert
g	converted to grams
kg	converted to kilograms
mt	converted to metric tons

Lines 10-80. Prints the introduction.

Lines 90-140. Displays the menu of conversions.

Lines 150-170. Asks for the units, sends the program to the correct subroutines.

Lines 180-210. Asks for the number of ounces which you are converting, and then goes to the conversion routine (since your measure is already in ounces, jumps in at line 310).

Lines 220-260. Asks for the number of pounds which you are converting, converts to ounces, and goes to the conversion routine (lines 310-420).

Lines 270-300. Asks for the number of tons which you are converting, converts to ounces, and goes to the conversion routine (lines 310-420).

Lines 310-350. Converts the ounces in your measure to grams, kilograms, and metric tons. (Notice how they only differ by factors of 10.)

Lines 360-420. Displays the converted measure, and ends.

Sample Runs

```
    ENGLISH/METRIC CONVERTER


THIS PROGRAM WILL CONVERT AN
ENGLISH UNIT OF LENGTH TO A
METRIC UNIT OF LENGTH.

WHAT UNIT OF LENGTH ARE YOU
CONVERTING FROM?

1.  INCHES
2.  FEET
3.  YARDS
4.  MILES
?1

HOW MANY INCHES ARE YOU
CONVERTING?
?2

THAT IS:
50.8 MILLIMETERS
5.08 CENTIMETERS
.508 DECIMETERS
.0508 METERS
5.08E-05 KILOMETERS

    ENGLISH/METRIC CONVERTER


THIS PROGRAM WILL CONVERT AN
ENGLISH UNIT OF VOLUME TO A
METRIC UNIT OF VOLUME.

WHAT UNIT OF VOLUME ARE YOU
CONVERTING FROM?
```



```
1.  OUNCES
2.  PINTS
3.  QUARTS
4.  GALLONS
?1

HOW MANY OUNCES ARE YOU
CONVERTING?
?15


THAT IS:
443.603 MILLILITERS
4.436 DECILITERS
.444 LITERS

    ENGLISH/METRIC CONVERTER


THIS PROGRAM WILL CONVERT AN
ENGLISH UNIT OF WEIGHT TO A
METRIC UNIT OF WEIGHT.

WHAT UNIT OF WEIGHT ARE YOU
CONVERTING FROM?

1.  OUNCES
2.  POUNDS
3.  TONS
?2

HOW MANY POUNDS ARE YOU
CONVERTING?
?55


THAT IS:
24948 GRAMS
24.948 KILOGRAMS
.028 METRIC TONS
```

Glossary

Glossary

address—A number that stands for a location in the computer's memory where a piece of information is stored.

air pressure—The force exerted by the weight of the atmosphere from the point of measurement.

assembler—A part of the computer that takes the information inputted and converts it into a form that the computer can understand.

asteroid—A small planet-like body that revolves around the sun. Also known as a planetoid or a minor planet.

asteroid belt—The region between Mars and Jupiter where most of the asteroids are found.

astronomy—The study of planets, stars, solar systems, galaxies, and the structure of the universe.

atom—One of the smallest particles of matter.

barometer—An instrument which measures the atmospheric pressure.

BASIC—(Beginner's All-purpose Symbolic Instructional Code) A computer language developed at Dartmouth College in 1963. It is a descendant of FORTRAN, and is unlike most other languages, but it has become the most popular for personal computers due to its stepwise instruction format.

binary—A system of numbering where there are only two numerals, 0 and 1. Also known as base 2, each digit in a binary number represents a power of two. This is how an electronic computer does its counting, as it can only deal with on or off. The computer translates each character of information into a string of binary numbers.

biology—The scientific study of life and living things.
bit—A binary digit, either a 0 or a 1. This is the smallest piece of information that a computer can read. These can be put together to make bigger pieces of information. See byte.
byte—A collection of bits. Most microcomputers have eight bits per byte. Each byte stands for a character (letter, number, or symbol). Therefore, it generally takes more than one byte to make up a word. For example, it takes eight bytes to make up the word computer.

chemistry—The scientific study of the properties of and changes in matter.
circumference—The length of a circle or the distance around a sphere (great circle).
clepsydra—A water-driven clock.
clinometer—A hand-held instrument used for measuring vertical angles. It is often used for finding the height of tall and/or faraway objects.
coriolis effect—Objects that are in motion on the earth's surface tend to be deflected to the right in the northern hemisphere and deflected to the left in the southern hemisphere because of the earth's rotation.
cosmic rays—Very high energy nuclear particles and radiation from outer space.
crystal—A solid in which the atoms are arranged in a repeating pattern. It has an orderly internal structure that is bounded by smooth planar faces.

deciduous—A name given to trees and shrubs that lose their leaves in the fall.

eclipse—An object casting a shadow on another object partially blocking or totally blocking the visibility of that object.
electromagnetic energy—The energy associated with electric and magnetic fields.
electromagnetic radiation—(EMR) Electromagnetic waves and electromagnetic energy.
electromagnetic spectrum—Invisible and visible energy waves ranging from very short wavelengths (gamma rays) to very long wavelengths (radio waves) that can travel at 186,282 mph (the speed of light) in a vacuum.
electromagnetic waves—Waves that are produced by a disturbance from an electrical charge. Far from the charge, the waves are made up of electric and magnetic fields that move at the speed of light.
ellipse—An oval shape. Planets move around the sun in an elliptical orbit.

escape velocity—The velocity at which an object can escape or leave the gravitational field of a planet.
evergreen—Trees that keep a set of leaves throughout the year.

fluorescent—An object that gives off or emits light when it is exposed to a form of electromagnetic radiation.
fossil—A cast, track, impression, or body part of an animal or plant that has been preserved in rock.
frequency—The number of vibrations of any given wave or particle in a unit of time.

gamma rays—High energy, high frequency waves which are shorter than x-rays.
geology—The scientific study of the earth, its composition, processes, features, and history.
geometry—The branch of mathematics which deals with the properties and relations of lines, angles, surfaces, and solids.
geosynchronous orbit—An orbit a satellite travels at a particular altitude and velocity in order to stay above a constant point over the earth's surface.
graphics—A display method on a computer screen or paper that produces graphs, charts, diagrams, symbols, and shapes (pictures).
gravitational field—The region in space in which an object or particle would experience a gravitational pull. The gravitational pull increases as the mass of the object increases. The gravitational force also increases as the distance between the objects decreases.
gravity—An attraction that occurs between all objects in the universe.

habitat—The place where a living organism lives and its surroundings.
hexadecimal—Base sixteen (16) numbering system. The digits are 0 1 2 3 4 5 6 7 8 9 A B C D E F. Most computers use this in their assembly languages. This is often referred to as hex.
humidity—The measure of the water vapor content of the air.

igneous—A rock formed by the rapid or slow cooling and solidification of magma.
infrared—Invisible electromagnetic waves that have a longer wavelength than visible light but are shorter than radio waves. They are used to dry and warm things.

mass—The measure of an object's ability to resist force.

matter—Anything that has mass and takes up space.
metamorphic—A rock formed when heat, pressure, and/or chemical action effects a preexisting rock altering it.
meteorologist—A scientist who studies the atmosphere and its phenomena.
microwaves—Invisible electromagnetic waves whose wavelength falls between infrared and radio waves. There are no specific boundaries that distinguish microwaves from infrared and radio waves.

neutron star—A massive star that is in the final stage of its "life cycle." It is made up of mostly neutrons and has a very strong gravitational attraction allowing very few particles to escape.

orbit—The oval path traced by a planet, meteorite, or satellite in the presence of a more massive body.
orbital period—The amount of time it takes a satellite to complete one revolution.

parallax—The apparent change in position of a faraway object when viewed from different points.
pendulum—A solid object that is suspended vertically from a fixed point and swings freely under the influence of gravity and momentum. It is often used to regulate movement in clocks.
pioneer—A series of space probes that were launched from earth by the United States whose mission was to investigate the outer solar system. Pioneer 11 left the solar system in 1983.
protractor—An instrument used for constructing and measuring angles.

radiation—Energy that is given off or emitted by waves through space or some medium.
relativity—A theory of physics based on the idea that the speed of light is constant. Everything else in the universe (time, length, and mass) depends on the motion of the observer.
resonate—To vibrate in harmony.
rock—An aggregate of two (2) or more minerals.

satellite—Generally, any small body that orbits a planet or star.
scientific notation—A system that makes the use of very large or very small numbers more convenient.
sedimentary—A type of rock formed by the compaction of loose sediment or by precipitation (usually in sea water).
species—Organisms that have many characteristics in common and can breed amongst themselves to produce fertile offspring. For example, a horse and a donkey are different species because their offspring (mule) is not capable of reproducing.

sputnik—The first artificial satellite launched from earth by the Soviets in 1957. It beeped for 21 days and extended man's horizon to space.

Telstar—The first communications satellite to transmit television programs overseas launched by the United States in 1962. It revolutionized intercontinental communications.
theory—General explanations based on scientific laws.
time dilation effect—According to the Special Theory of Relativity, the closer a traveler is to the speed of light, the slower time on board the traveler's spaceship would appear to pass to an external observer.
trigonometry—The branch of mathematics that deals with determining the sides and angles of triangles. It is often used in engineering, surveying, navigation, electricity, construction, etc.

ultraviolet radiation—Invisible electromagnetic waves that have wavelengths shorter than visible light. It is often used to kill germs in hospitals and to increase vitamin D content in some foods. Ultraviolet radiation is present in sunlight and upon its contact with your body, it produces vitamin D in your skin. Too much exposure to UV radiation can be harmful and cause skin cancer.

velocity—Speed in a direction.
visible light—Visible electromagnetic waves whose wavelengths are longer than ultraviolet and shorter than infrared waves. When visible light is passed through a prism, it produces bands of colors.
volume—The amount of space that matter takes up.
voyager—Space probes launched in 1977 by the United States whose mission was to investigate the outer solar system. Among the phenomena that the Voyagers discovered were the volcanoes of IO, the icy surface of Europa (in the Jupiter system), and the shepherd moons of Saturn's rings.

wave—A rhythmic disturbance which moves through or over space or matter.
wavelength—The linear length of one wave.
weight—Force due to the effect of gravity.
white dwarf star—A less massive star that is in the final stage of its life cycle.

x-rays—Electromagnetic waves whose wavelengths are shorter than those of ultraviolet radiation and longer than those of gamma radiation. X-rays were discovered by the German physicist W. K. Roentgen.

Index

Index

A

Addresses, 107
Alexandria, 72, 73
Alphabet, 130
Angle, 117
Angle, elevation, 118
Ant, 81, 82, 84
Asteroid belt, 8
Asteroids, 25
Astronomer, 123
Astronomy, 7, 145
Atmosphere, 134

B

Barometer, 116
Basalt, 48
Base, power of, 110
Base 2, 109
Base 3, 109
Base 10, 107, 125, 160
Base 16, 110
BASIC, 110, 135
Billion, 123, 126
Binary, 109, 110
Biology, 100-102
Biosynthesis, 101
Bird, 87
Birth, 97
Bits, 107
Black hole, 28, 29
Botanists, 91
Bradley, James, 143
Bytes, 107

C

Carbon, 82
Caribbean Islands, 69
Catalog program, 6
Cedars, 90
Census Bureau, 131
Chemistry, 134
Chinese, 27, 107
Chronobiology, 102
Chronometer, 102
Clepsydras, 152
Climate, 41
Clinometer, 119
Clocks, 151
Cloud chart, 56
Cloud identification, 54
Clouds, 63
Coal, 49
Columbus, Christopher, 68, 71
Compass, 45, 65
Computer, 107, 109, 130
Computer science, 107
Conglomerate, 48
Constant, 117
Coriolois effect, 62
Cosmic radiation, 140

D

Daedalus, 32
Darwin, 101
Datapacks, 1
Death, 97
Decay, 96
Deciduous, 89, 91
Decimal, 110
Dilation, 147
DNA, 101

E

Earth, 41, 81
Earth, flat, 68, 71
Earth science, 41
Earth's circumference, measuring the, 68, 74
Earth's curvature, 70
Editing programs, 2
Einstein, Albert, 143, 145, 146
Electromagnetic radiation, 139
Elevation angle, 118
EMR, 139-141
Encyclopedia, 10, 32, 38, 43, 98
Energy, 139, 145
Engineering, 134

English system of measurement, 158, 160
Entering programs, 4
Equator, 72
Eratosthenes, 71, 72, 73, 76
Escape velocity, 17, 27, 28
Escape velocity equation, 28
Estimation, 124
Evergreen key, 89
Evergreens, 90
Explanations, 6
Extinct, 96
Eye, 141

F

Fauna, 81
Flat Earth Society, 68
Flora, 81
Fluorescent bulb, 142
Foot, 157
Force equation, 23, 28
Forecast, 64
Forecast, twelve-hour weather, 60
FORTRAN, 110
Fossils, 49
French, 107
Frequency, 139, 143

G

Gabbro, 48
Galilei, Galileo, 152
Game comtrollers, 1
Gamma rays, 140
Generation, 97
Geology, 41
Geometry, 74
Geosynchronous, 17
Getting started, 1
Googol, 123
Granite, 48
Graphics, 111
Gravity, 17, 21, 22, 23, 27, 153
Greek, 107
Greek alphabet, 130
Greek mythology, 32, 68
Growth, 96, 97

H

Height, 116, 117, 118
Hemlocks, 90
Hertz, 143
Hertz, Heinrich, 143
Hexadecimal, 110
Hollerith, Herman, 131
Hollerith punch cards, 131
Honeybee, 84, 85, 86
Humidity, 55, 63
Hurricanes, 61
Hydrogen, 82

I

Igneous, 47
Inches, 158
Infrared radiation, 140
Insects, 82, 87
Isabella, Queen of Portugal, 68

J

Jet, 85

K

Kasner, Edward, 123
Kemeny, John, 135
Kepler, Johannes, 17
Keyboard, 1

L

Laboratories, 134
Language, 107
Larch, 91
Leaf, 90, 91
Length, 147, 148, 153
Life science, 81
Light, 140-141
Light, speed of, 143, 145, 146
Limestone, 49
Linnaeus, 101
Loading programs, 3
Logic, 68, 109
Logo, 110
Lorentz Transformation, 147

M

Magma, 47
Manuals, 1
Mass, 24, 25, 28, 139, 147-148, 153
Mathematicians, 124
Mathematics, 107
Measurement, 157
Measurement, direct method of, 34
Measurement, indirect method of, 35
Medicine, 101, 134
Memory console, 1
Mendel, 101
Metabolism, 101
Metamorphic, 49
Meteorologists, 54, 55
Metric conversions, 157
Metric system of measurement, 160
Microwaves, 140
Million, 123, 125
Model, 10, 42
Muscle fibers, 83
Muscles, 82
Music, 111

N

National Weather Service, 64
Needles, 90, 91
Neutron star, 28
Newton, Sir Isaac, 22
Nitrogen, 82
Nuclear power, 140
Numbers on the screen, 5

O

Obsidian, 48
Oil, 49
Orbits, 10, 17

P

Parallax, 35
Pascal, 110
Pendulums, 151, 152
Period, 17
Physicist, 123
Physics, 139, 145
Pine trees, 90
Pinhole camera, 37
Pioneer, 15
Place value, 111
Planetary diameters, 7
Planetary distances, 7
Planets, orbits of, 10
Plants, 142
Population, 96
Power, 125-126
Power of 10, 125
Power of the base, 110
Precipitation patterns, 62
Prefixes, 160
Pressure, 55, 62
Pressure, region of low, 62
Printer, 1
Probes, 14-15
Program, alpha beta, 136
Program, alphabetize words, 132
Program, base converter, 112
Program, biology matching quiz, 103
Program, catalog, 6
Program, computing satellite orbital periods, 18
Program, distance to the sun, 38
Program, escape velocity and gravitational pull, 30
Program, evergreen key, 92
Program, forecast the weather, 66
Program, frequency and wavelength, 143
Program, geological/universal time clock, 44
Program, height finder, 120
Program, identify that rock, 50
Program, identify the clouds, 58
Program, mass change near light speed, 150
Program, measure the earth's circumference, 76
Program, metric length converter, 161
Program, metric volume converter, 162
Program, metric weight converter, 164
Program, pendulums, 154
Program, population simulation, 98
Program, scaled distances and diameters for the planets, 12
Program, scientific notation, 127
Program, supersonic bees, 87
Program, weights on different planets, 25
Protractor, 119
Pumice, 47
Punch cards, 131

R

Radiation, 41, 140
Radio waves, 140, 141
Relativity, 146
Remarks, 6
Rhyolite, 48
Right triangles, 117
River beds, 134
Robots, 8
Rocket, 15, 27, 146
Rock identification, 46
Romans, 102
Romer, Oleas, 143
Running programs, 2

S

Sample run for alpha beta, 137
Sample run for alphabetize words, 133
Sample run for biology matching quiz, 105
Sample run for computing satellite orbital periods, 20
Sample run for escape velocity and gravitational pull, 30
Sample run for evergreen key, 93
Sample run for forecast the weather, 68
Sample run for frequency and wavelength, 145
Sample run for geological/universal time clock, 45
Sample run for height finder, 122
Sample run for identify that rock, 53
Sample run for identify the clouds, 58
Sample run for mass change near light speed, 151
Sample run for measure the earth's circumference, 79
Sample run for metric conversions, 166
Sample run for pendulums, 156
Sample run for population simulation, 99
Sample run for scaled distances and diameters for the planets, 13
Sample run for scientific notation, 129
Sample run for supersonic bees, 89
Sample run for the base converter, 115
Sample run for the distance to the sun, 40
Sample run for weights on different planets, 26
Sandstone, 48
Satellite orbital periods, 14
Satellites, 7, 14-15, 57
Saving programs, 3
Scientific notation, 123, 126, 127
Scientists, 134
Screen formating, 6
Sedimentary, 48
Shale, 48
SmartBASIC, 1, 4
SmartWRITER, 4
Snail, 87
Solar system, 10, 41, 81
Spacecraft, 15
Space science, 7
Space shuttle, 8, 130
Species, 101
Speed of light, 143, 145, 146
Speed of sound, 85
Spruces, 90
Sputnik, 15
Star, 28, 29, 36
Sugar, 90
Sun, distance to, 32
Syene, Eygpt, 72, 73
Syntax, rules of, 5

T

Tamarack, 91
Tangents, 120
Tape cassettes, 1
Telescopes, 36
Telstar, 15
Temperature, 55, 63-64, 95
Thermometer, 65
Time, 147, 148
Tornadoes, 61
Triangles, right, 117
Trigonometry, 36, 117
Trillion, 123
Tropical storms, 61
Tv interface, 1
Tv signals, 140, 141

U

Ultraviolet light, 140

V

Velocity, 27, 28
Volcano, 46, 47, 134
Voyager, 15

W

Wavelength, 139, 143
Weather, 15, 41, 54
Weather balloons, 56
Weather forecast, 60
Weather patterns, 62
Weather prediction, 58, 60
Weather report, 57
Weight on different planets, 21
West Indies, 69
White dwarf, 28
Wind direction, 64-65

X

X-rays, 140, 141

Edited by Brint Rutherford

Discovering Science on Your ADAM™ with

by Talcott Mountain Science Center

Here's a series of 25 ready-to-run science programs for use on the ADAM computer from Coleco that makes it possible for students of all ages to find new things, see new places, and meet new people! It's a guidebook that's sure to fire any student's imagination and to make science a favorite subject!

Using simple programming techniques, students can tour the solar system . . . forecast weather . . . identify rocks and other geological specimens . . . design a pendulum . . . not to mention the opportunity they'll have to rub shoulders with Newton, Darwin, Kepler, Galileo, Einstein, and other scientific geniuses!

The best part about this programming guide is that youngsters can have fun learning as they embark on adventures in the various areas of scientific achievement. Written in interesting and lighthearted style, the book supplies a ready-to-run program in ADAM BASIC for each scientific adventure. In some cases, there are additional projects to go with the program (directions for building a model of the solar system to go with the program adventure on planet distances and diameters, for example). Plus, the authors have included step-by-step instructions on using the ADAM, editing programs, saving programs, loading programs, rules of syntax, screen formatting, and more.

Among the science areas covered are adventures in astronomy and space science . . . adventures in earth science including geology and weather forecasting adventures in life science . . . adventures in mathematics and computer science . . . and adventures in physics.

Designed for individual student use or as a supplementary text for use in middle school or junior high science classes, this guide provides an ideal way to create student interest in science . . . and in computer programming!

The Talcott Mountain Science Center in Avon, Connecticut is a unique, private, non-profit educational and research facility committed to a hands-on approach to science teaching and learning. Serving students ranging from kindergarten to university level from public, private, and parochial schools in over 100 Connecticut communities, the Center also provides instruction for teachers, administrators, and community members and is working on applied research for business, industry, communities, and government agencies.

TAB TAB BOOKS Inc.
Blue Ridge Summit, Pa. 17214

ISBN 0-8306-1780-9

965-0884