DOCUMENT RESUME

ED 231 636 SE 042 072

TITLE	Input-Output. Teacher's Guide.
INSTITUTION	San Francisco State Univ., Calif.
SPONS AGENCY	National Inst. of Education (ED), Washington, DC.; National Science Foundation, Washington, D.C.
PUB DATE	81
GRANT	SED-80-12465
NOTE	63p.; For related documents, see SE 042 071-079. Pages listing computer program code may not reproduce well.
PUB TYPE	Guides - Classroom Use - Guides (For Teachers) (052) -- Audiovisual Materials (100)
EDRS PRICE	MF01/PC03 Plus Postage.
DESCRIPTORS	*Computer Oriented Programs; Elementary Secondary Education; Learning Activities; Mathematical Concepts; Mathematics Curriculum; *Mathematics Education; *Mathematics Instruction; *Microcomputers; Middle Schools; *Pattern Recognition; Problem Solving; Programing; Programing Languages; Teaching Guides; Units of Study; Worksheets
IDENTIFIERS	*Easy Speak (Computer Program); Math Network Curriculum Project; *Wizard (Computer Program)

ABSTRACT

This document is the first of seven units developed by the Math Network Curriculum Project. Each unit, designed to be a 2-week module, is a teacher's guide which includes detailed directions along with the courseware and software needed. Teacher intervention in the non-computer activities that begin each unit is required, and the consistent use of small-group instruction makes the units usable in a standard classroom if two microcomputers are present. The Input-Output Unit develops students' abilities to recognize and describe patterns and introduces the notion of expressing mathematical concepts in a symbolic language, Easy Speak. The experience of using Easy Speak prepares students to use algebra to express numerical ideas. Practice is provided on simple input-output machines, many of which grow out of activities with manipulative materials, and moves to machines with two-part rules. Printed copies of the code for the Wizard and Easy Speak computer programs are included. Wizard reinforces the idea of conditional statements by having students create their own and interact with the computer. Easy Speak, used with Easy Speak Summary, permits students working in pairs to create mystery machines for pairs of numbers. They can also try mystery machines created by students in other classes. Both programs were developed for use on a Commodore PET Computer with at least 16K of RAM using 4.0 BASIC. (MNS)

Input-Output

Teacher's Guide

Math Network Curriculum Project

San Francisco State University

Funded by the National Science Foundation and the National Institute of Education
Project No. SED 8012465

INPUT - OUTPUT UNIT

TEACHER'S GUIDE

Math Network Curriculum Project

TABLE OF CONTENTS

OVERVIEW

The ability to recognize patterns is a key to mathematical thinking. Patterns are plentiful in almost every area of mathematics and science. Searching for these patterns is a way of thinking that is essential for approaching mathematics at any level. Mary Barratta-Lorton points out this significance for learning:

> "Looking for patterns trains the mind to search out and discover similarities that bind seemingly unrelated information together as a whole...A child who expects things to "make sense" looks for the sense in things and from this sense develops an understanding. A child who does not see patterns often does not expect things to make sense and sees all events as discrete, separate and unrelated."

In this unit, students will search for and create elementary functions.

The basis of this unit is in the numerical patterns found in the Input-Output format. Student activities are organized around Input-Output Machines and a fantasy of ficticious Wizards. Experiments using concrete materials are used to provide a "real world" experience that parallels the computer-based activities. The computer activities culminate in the students' use of the language EASY SPEAK to create functions of their own for other students to explore.

In addition to the fantasy, this unit depends on cooperative learning in groups of four to support student learning. This cooperative mode of learning can also ease the teacher's task by reducing the need for teacher contact with individual students and by encouraging students to be more self-reliant.

The combination of work with the number patterns of elementary functions and the use of EASY SPEAK to create operating rules for more complex functions can give students a working understanding of fundamental concepts from Algebra and at the same time provide them with reasonable practice in arithmetic skills.

How to Use this Guide

The activities of this unit are designed to be presented in sequence. Each activity is presented with the following categories:

FORMAT
TIME
MATERIALS
BACKGROUND
PURPOSE
TEACHER PREPARATION
ACTIVITY

The ACTIVITY section describes in detail what you and your students will be doing. Along with specific teaching instructions and directions for student activities, ideas for classroom management are included. A significant classroom organization idea of this section is the use of groups of four. This entails:

Randomly putting students into groups of four each. You might hand out playing cards (or numbered slips of paper) or have students count off to create the groups as called for in this guide.

Arranging classroom seating by pushing desks into groups.

Circulating among the groups during the activities and offering help when necessary. Avoid interrupting the entire class while they are working in groups.

Discussing operating rules. Cooperative group work may be new to many students. Help them understand the "team" aspect of their assignments. Require that they ask questions of one another before they ask you.

Summary of Activities and Materials

1. THE FABLE OF THE WIZARDS (1/2 day)
A tongue-in-cheek fable to introduce the input-output idea.

OVERHEAD PROJECTOR
MYSTERY TOWER transparency

2. INPUT-OUTPUT MACHINES (1 day)
A review of the input-output idea. Students will create their own elementary functions. The group process is introduced.

OVERHEAD PROJECTOR
MYSTERY TOWER transparency
INPUT-OUTPUT MACHINE transparency
OTHER MACHINES worksheet
DISCUSSION WORKSHEET

3. MYSTERY MACHINES (1 or 2 days)
The idea of conditional statements for input-output machines is introduced.

OVERHEAD PROJECTOR
INPUT-OUTPUT MACHINE transparency
MYSTERY MACHINES worksheet
EXAMPLES OF CONDITIONALS worksheet
WORKSHEET A
WORKSHEET B

4. WIZARD (1 day)
This activity reinforces the idea of conditional statements by having students create their own and begin work on the computer.

PET computers
WIZARD program
OTHER MACHINES worksheet

5. WIZARD'S WORKSHOP (2 days)
This activity provides hands-on experiments for students to develop elementary functions.

PET computers
WIZARD program
WIZARD'S WORKSHOP worksheets
WIZARD'S WORKSHOP experiment cards (8)
CUISENAIRE RODS
CUBES
BEANS OR OTHER SMALL COUNTERS
SMALL PAPER CUPS (optional)

6. THE HUMAN COMPUTER (1 or 2 days)

An activity to give students experience in writing and checking EASY SPEAK statements.

PET computer
EASY SPEAK program
EASY SPEAK SUMMARY
HUMAN COMPUTER WORKSHEET

7. EASY SPEAK (3 or 4 days of individual work)
The final computer activity of this unit. Students work in pairs to create EASY SPEAK programs.

PET computer
EASY SPEAK program
EASY SPEAK worksheet

THE FABLE OF THE WIZARDS

Format
Whole class

Time
20-30 minutes at end of period

Materials Needed
overhead projector
MYSTERY TOWER transparency

Background

The idea of a fable creates a non-threatening fanciful situation and releases students' creativity to become involved in the tale. This fable is to be told tongue-in-cheek, and classroom experiences with the idea indicate that even the most skeptical student gives in to a playful involvement. The wizard theme is carried throughout the unit, and if you keep up the spirit of the idea, students will accept the playful tone quite readily. The end-of-the- period timing gives students time to internalize the input-output ideas before actually working with it on their own.

Purpose

- To introduce students to the input-output concept;
- to have students verbalize the rules for simple functions in English.

Teacher Preparation

Before class, prepare or have ready the overhead transparency of the MYSTERY TOWER, which you will project directly onto the chalkboard.

Activity

Settle the class, asking them to put everything away because you are going to read a fable.

The Fable

Long ago in Transylvania, the home of many mysterious creatures and unusual people, there existed a society of wizards. In those days wizards did many extraordinary things, calling on the help of magic powers. We now know that among these people were included what we call "math" wizards. Most of their wizardry

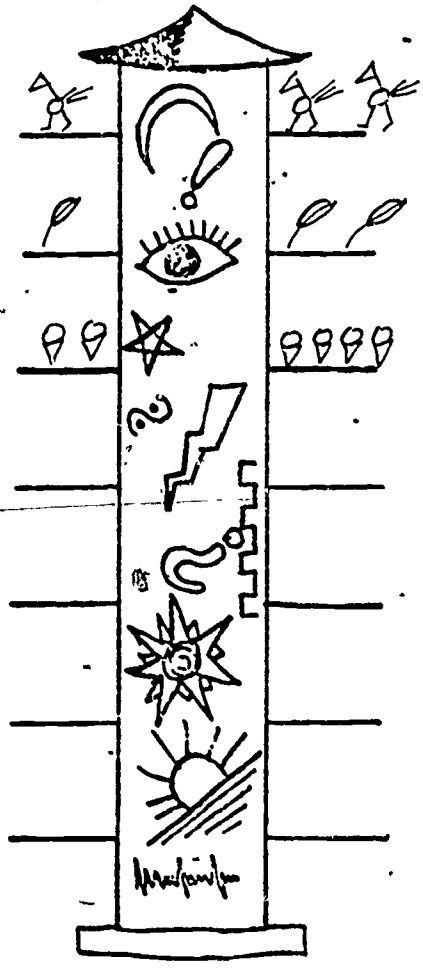
really came from an understanding of mathematics, but, no matter, it seemed like magic to others. It turns out that every math teacher is a bit of a Wizard, a math wizard, that is. One bit of wizardry I will reveal to you so that each of you has the opportunity to become a part of this society.

(Show Mystery Tower projected on chalkboard. You will be writing on the board image of the tower.)

Here is an exact replica of the Wizards' Mystery Tower which was located in the Transylvanian forest. At that time, the secret of this tower was known only to the society of wizards, but I'm going to show you how it works.

On stormy days when lightning and thunder filled the skies, Wizards, like us, would gather to watch what the tower would do.

Each time a bolt of lightning struck the tower, it would take on a whole new function. Here is one example. See if you can figure out what the tower is doing.



(Here follows a series of inputs that you should take the time for all children to understand. Caution the class about calling out or telling answers because each person needs to develop his or her own skills as a Wizard.)

If a bird 
would land on the left side of the tower, two birds would appear on the right.

If a leaf 
...then two leaves ...

(Sketches of these and one or two other examples are quickly drawn by you on the appropriate spaces on the tower, e.g., hat  , nickel . Ask the class what would happen if a Volkswagen  were put on the left.

As often happens in Transylvania, lightning struck the tower and the magic symbols would flash and glow. The wizards watched carefully for those opportunities, as unlocking the secrets of the symbols always added to their mathematical wizardry.

(Erase the first example from the board image.)

Here's the kind of information they were known to get from such a flash.

(At this point, fill in the 3 or 4 pairs of spaces on the tower with a very simple function (like one that adds 3). Allow the students to ask about it.

Sometimes wizards would test themselves (and each other) just to keep their skills honed. They'd make lightning strike to change the function of the tower, and then try out symbols, trying to guess the function in as few times as possible. This kept them busy on slow days. Try it. Give me a number, and I'll tell you what output the Tower would give.

The third example could be a combination of the first two (doubles and adds 3)

Now the fable is left behind, but the questioning continues for a new function. Have students suggest an input number, and give them the output. See if students can give the output. The functions you choose should be kept simple.

e.g., Subtract 1
Double and add 2
Double and subtract 1
Triple and add five.

When most students have unraveled each function, have one student describe what the tower does.

TEACHING NOTES

Experts - Some students will see the relationships much faster than others. Ask them not to describe the tower, but designate them as "experts" or "checkers", and check guesses from the class with those who believe they "get it".

Answers - Be careful to keep those who know from calling out the rule. Everyone needs to practice their wizard skills.

Repetition - Ask several students about the same input. Accept answers without judgment. Merely ask: Does anyone have another thought? Does anyone agree? When all who wish to have expressed themselves, record the correct number.

Hints - Some patterns become more apparent when the input is rearranged in numerical order. Students do not generally give inputs in numerical order. After they've become familiar with the activity, suggest that analyzing the numbers in order reveals useful patterns. Rearrange one to demonstrate.

INPUT-OUTPUT MACHINES

Format
Whole class and groups of four

Time
45-50 minutes

Materials Needed
transparency of MYSTERY TOWER
overhead projector
transparency of INPUT-OUTPUT MACHINE
OTHER MACHINES worksheet
DISCUSSION WORKSHEET

Note: OTHER MACHINES is a blank worksheet on which you or your students can create Input-Output machines.

Background

The input-output idea has been presented, and this day is used for review and to have students use the idea in a cooperative groups of four setting. Careful introduction of procedures for setting up and working in groups of four is an important part of this activity.

Purpose

- To reinforce the input-output concept.
- To establish cooperative group work.

Teacher Preparation

Make copies of the OTHER MACHINES ditto with the tables filled in with simple functions for students to solve. Some examples: multiply by 5 and add 2; multiply by 3 and subtract 2; subtract input from 20 (this will be harder). You may also include one that squares the input and adds 1. You might also choose to leave one blank and instruct the students to create one of their own.

You should feel free to create additional machines for your students. At times you could present the inputs in sequence; other times, mixed up; other times, an odd assortment of inputs. You should try to use numbers that are easy to work with, either small numbers or larger numbers that help students with the underlying pattern (such as 100 or 50 or some other multiple of 10). Be sure you have given at least five INPUTS for each machine and do not use division. Repetition of examples you may have done earlier is acceptable.

Review the ideas in the unit overview on groups of four.

Activity

Review

Divide the class into groups of four. Explain group procedures to the class. Project the MYSTERY TOWER on the board and have students verbalize what they recall about the activity. Accept all comments but try to draw out:

1. You input a number and another one comes out.
2. It does the same thing each time.

By way of review, do another simple example such as double and add two. Write in the input values from 1 to 7. Then write in the output values one at a time letting students participate as before. Ask about other inputs such as 20, 13, 10. Repeat an earlier input to emphasize that the order does not affect the resulting output. Reinforce the idea that the MYSTERY TOWER does the same thing to the input each time.

INPUT	OUTPUT

Turn off the overhead **before** erasing the numbers and explain that from now on, we will use a simpler diagram (called a table) to record the Input and Output. Draw a T shape to separate the two sets of numbers that are still on the board.

Using this same T shape (the table), go through another simple example (e.g., multiply by 3 and subtract 1), drawing from students an input number and recording on the table. Maximize student involvement and reinforce the no-telling rule. Have someone verbalize the rule at the conclusion.

Group of Four Activity

In groups, each student writes down an easy rule for others in the group to discover. Take turns. Whoever invented the rule makes a table to record guesses and outputs. When the guessers figure out the rule, one records it below the table. When all have tried this, have each group decide on a rule they'd like to present to the rest of the class. Each group chooses a spokesperson to present the rule for others to discover.

After each group has presented a rule, show class the overhead of the INPUT-OUTPUT MACHINE. This is the modern version of the MYSTERY TOWER.

Hand out the worksheet you have prepared and have students work individually on solving them either in class or as a homework assignment.

You may want or need to provide even more examples for students to try, either as a whole class or in small groups or individually. The students should be encouraged to see these as a challenge and should feel comfortable working on them. If your class is having difficulty, you may want to use the OTHER MACHINES worksheet to give the rules and have them fill in the tables for selected inputs.

Discussion

Spend some time having the class give each other hints that they have discovered that help guess the rules. This can occur after the OTHER MACHINES worksheet if the class is doing well or as a follow-up to the DISCUSSION WORKSHEET if you choose to use it as a summary task.

Some points that may arise in the discussion are: put the terms in order; look for patterns; try a lot of rules for the same INPUT and check each one on other inputs. Let students express their ideas in their own terms. Try to value each contribution equally.

MYSTERY MACHINES

Format
Whole class and groups of four

Time
two class periods

Materials Needed
- overhead projector
- INPUT-OUTPUT MACHINE transparency
- MYSTERY MACHINES worksheet
- EASY SPEAK summary for reference
- EXAMPLES OF CONDITIONALS worksheet
- WORKSHEET A
- WORKSHEET B

Background

The introduction in this activity of MYSTERY MACHINES is intended to be a prelude to the students' understanding of EASY SPEAK. Easy Speak is a symbolic language that expresses numerical and algebraic ideas in a form understood by several computer programs in this curriculum. The Strategies Unit and the Business Unit follow up what the students learn here. Easy Speak "sentences" are a sequence of statements that create input-output machines and are written in the form:

```
CONDITION:
OUTPUT=
```

The "CONDITION" portion will usually be a statement involving the input to the machine and the "OUTPUT" portion will describe the resulting output.

Examples:

```
1.  CONDITION:INPUT<6
    OUTPUT=INPUT
2.  CONDITION:INPUT>=6
    OUTPUT=10
```

```
1.  CONDITION:EVEN(INPUT)
    OUTPUT=INPUT+1
2.  CONDITION:ODD(INPUT)
    OUTPUT=20-INPUT
```

```
1.  CONDITION:FACT(3,INPUT)
    OUTPUT=11
2.  CONDITION:NOT(FACT(3,INPUT))
    OUTPUT=2*INPUT
```

The richness of EASY SPEAK provides a medium for the students' understanding and expression of many mathematical ideas, several of which appear in this activity.

Purpose

- To introduce conditional statements using < (less than), > (greater than), ≤ (less than or equal to), ≥ (greater than or equal to), <> (not equal to), EVEN, ODD, FACT, and NOT.

Teacher Preparation

Run off copies of the MYSTERY MACHINES worksheet. Run off copies of the EXAMPLES OF CONDITIONALS worksheet.

Review the descriptions of < (less than), > (greater than), ≤ (less than or equal to), [>] (greater than or equal to), <> (not equal to), EVEN, ODD, FACT, and NOT in your EASY SPEAK summary.

Activity

INPUT	OUTPUT
1	1
2	2
3	3
4	4
5	5
6	10
7	10
8	10
9	10

Divide the class into new groups of four. Quickly go over the worksheet from last time.

Project the INPUT-OUTPUT MACHINE transparency on the board (or draw a facsimile).

Suggest to the class that today's more modern machines can do what the MYSTERY TOWER did and much more. Copy the following tables one at a time onto the projection of the transparency. You may want to present the output values one at a time to allow students more opportunity to discover the difference between these conditional machines and others they have done previously. Students should be encouraged to express the patterns and conditions they discover in their own terms. You may want to create additional examples of your own for each of the three types presented in the diagram. An EASY SPEAK form for these tables is in the "Background" section.

INPUT	OUTPUT
1	19
2	3
3	17
4	5
5	15
6	7
7	13
8	9
9	11

Take whatever time you need to cover each type of example. students should see a complete and orderly table in order to detect the patterns.

Tell class that fortunately the conditions EVEN, ODD, FACT (and several other EASY SPEAK conditions) require the use of NATURAL NUMBERS, which are the numbers we use to count the people in this room (1, 2, 3, etc...) .

INPUT	OUTPUT
1	2
2	4
3	11
4	8
5	10
6	11
7	14
8	16
9	11

You may want to try more than one of each of the three types (EVEN, FACT, <), but keep the OUTPUT very simple. Descriptions from students will be in English, and you should begin at this point using the EASY SPEAK notation so they will see it written before they have to write it.

Try other examples of your own. Have the students suggest the inputs one at a time. Ask how many

inputs they think they will need to figure out your mystery machine. There is no right answer to this question.

Review with them if necessary, the meaning of EVEN, ODD, FACT, <, ≤, >, ≥.

Group of Four Activity

You will give them the EXAMPLES OF CONDITIONALS worksheet, and their task is to match a description with a table. There is one table and one discription that does not match. The group should agree completely on their choices.

Give them the MYSTERY MACHINES worksheet to complete for homework.

Worksheets A and B can be used for additional in-class work or homework as needed after students have completed the group activity.

Discussion

Discussion of what students tried and how they worked together in groups is critical to bring in the students who are having difficulty and to support strategies that other students have discovered.

The EXAMPLES OF CONDITIONALS worksheet is an excellent opportunity to discuss the syntax of the EASY SPEAK statements they have seen. In particular, it is a time to clarify the use of NOT as an EASY SPEAK statement.

Worksheet B provides the opportunity to seek out different descriptions. Table B on this worksheet is really ambiguous as to where the two different conditions take effect. All that you can tell is that somewhere around the numbers 8, 9, 10, 11 there is a change (e.g., <=8, or <9 or <10 or <11 will work as a condition on the smaller inputs). For the larger inputs, >=11, >10, >9, and >8 will satisfy the table.

WIZARD

Format
Whole class followed by groups of four. Some groups on computers.

Time
45-50 minutes. The computer part will extend into the next two days.

Materials Needed
OTHER MACHINES worksheet
WIZARD program
PET computers

Background

It is important to go over the MYSTERY MACHINES worksheet carefully to insure that students understand conditional input-output machines and the language used to express those conditions. Students need not be able to write EASY SPEAK statements at this point, even though some of them will. The main goal here is the understanding and recognition of conditional in-out tables. The second half of this period is for students to create their own conditional machines or to decode the machines presented to them by the WIZARD program. Students will earn wizard rankings and collect treasures for each machine successfully completed.

Individual students will be asked to select the level of difficulty, but the group as a whole earns a ranking and collects the treasures.

Some Notes on the Rules the Wizard Program Creates

The number requested for level of difficulty (1-4) sets one of four levels. The two-digit seed number produces unique rules, unique "locations", and ten possible treasures.

Level 1 and 2 use linear rules (y=A*INPUT+B) and rules involving constants. Level 2 also uses FACTOR as a condition. Level 3 adds the possibility of negative numbers in the rule. Level 4 incorporates the possibility of simple quadratic (second degree rules) (Y=A*INPUT↑2+B). Since the rules are created randomly, the higher numbers within a level are not necessarily in order of increasing difficulty (that is to say that seed number of 86 is not necessarily more complex than 26).

The two types of scoring are the treasures earned and a Wizard ranking: NOVICE WIZARD, JUNIOR WIZARD, WIZARD, MASTER WIZARD. The treasures are earned when a machine is solved. The Wizard ranking is based on the number of machines tried, the efficiency of getting the solution, and the levels attempted.

Purpose

- To further develop students' informal understanding and use of conditional statements using EVEN, ODD, FACT, < (less than), > (greater than), <> (not equal), ≤ (less than or equal to), and ≥ (greater than or equal to)

Teacher Preparation

Run off two copies of OTHER MACHINES worksheet for each student. Load WIZARD on both computers.

Activity

Have students sit together in their previous groups. Poll the groups about the MYSTERY MACHINES worksheet of the previous day. Which were easy? Which were hard? Try to draw out different descriptions for the same machine. In some cases, you may have to give another description to show that more than one answer is possible. Pay particular attention to the conditional statements necessary to specify the rule.

> Example: FOR MACHINE B, If the input is even, the output is double the input. If the input is odd, the output is eleven times the input. Another description might be if the input is even, the output equals input plus input. If the input is odd, the output is ten times the input plus the input.
>
> Machines A, E, and F are similar in that there is a single rule **possible** for all input. Try to develop the idea of "When does the rule work?" The sought-after response that matches the EASY SPEAK language is "ALWAYS".

Group of Four Activity

Hand out two copies of the OTHER MACHINES worksheet to each student. Have them label one, "COMPUTER MACHINES" and the other, "GROUP MACHINES". There are two assignments which are to be completed over the next few days.

Assignment 1. Each group is to work together to create four "interesting" machines that involve conditional statements. One of these will be shared with the class at a later time.

Assignment 2. Each group is to write descriptions and record the tables for the four most interesting machines that they solved from the WIZARD program. They should be sure to record the difficulty level and the seed number so that the rest of the class may try them.

Explain that the groups will be working together for the next two days. In that time, each group will have time scheduled on the computer to complete Assignment 2. Two groups should start on the computer assignment now and complete Assignment 1 tomorrow. Post a schedule of times. Each group will need about half a period of computer time. The computer activity for the remaining groups will continue while most of the class works on the next activity, WIZARD'S WORKSHOP.

WIZARD'S WORKSHOP

Format
Entire class working in modified groups of four

Time
Two class periods

Materials Needed
- 3 or 4 sets of Cuisenaire Rods
- 100 cubes
- 100 beans or small counters
- 10-20 small paper cups (optional)
- PET computers
- WIZARD program
- WIZARD'S WORKSHOP worksheets
- WIZARD'S WORKSHOP experiments

Note: There are eight different experiments.

Background

This activity presents eight experiments for students to create numerical patterns and discover the input-output rules. Six of the eight also provide a follow-up experiment, "FOR MASTER WIZARDS", which uses the same materials. The materials used are important to reinforce the input-output concept and give the students a firsthand experience in creating these "real world" input-output patterns. Most of the materials are readily available. The Cuisenaire Rods may be obtained from other teachers or from a local elementary school or resource center for the short time you will need them. The two days allotted for this activity gives time for the remaining groups to work with the WIZARD program.

Purpose

- To provide concrete experiences creating and solving simple functions.

Teacher Preparation

Run off back-to-back copies of WIZARD'S WORKSHOP worksheets. Two per student should suffice.

Run off 3 copies of each of the WIZARD'S WORKSHOP experiments. These pages are designed to allow you to ditto them onto 5x8 cards. Otherwise, you could run them off on regular paper, and then, mount each one on card stock.

Arrange the materials on a table or counter with the appropriate experiment cards nearby.

Activity

Have students sit together in their groups.

Remind the class of the schedule for the remaining groups to complete the WIZARD program. The groups that did Assignment 2 last time should complete Assignment 1 before going on to the experiments.

Describe the Wizard's Workshop process, showing them the experiment cards and the worksheets. Explain that they will be working in pairs from their groups and that they should procede at their own pace, being sure to record the results for each experiment. Students may have a tendency to speed through the experiments without checking their conclusions on larger input numbers. The criteria question for each experiment is marked with an asterisk and intended to check students' grasp of the underlying rule. You should circulate while the students work and watch for instances of premature conclusions. You may choose to present all eight experiments at once or hold back half of them to be introduced at the beginning of the next day. Students may not have time to complete every experiment.

Point out that some experiment cards have two activities: a regular experiment and a Master Wizard's Experiment. Each experiment has a criteria question that is starred (*), and that the same question applies to the experiment for Master Wizards at the bottom of most cards.

The experiments are designed to be self-explanatory, but it would be useful to have the titles listed on the chalkboard so students can decide which to do next while still seated.

Answer questions. Ask pairs to decide which experiment they will try first, then go to work. (What happens if not enough materials are available? Discuss that with the class.)

Circulate while students are working. Interrupt work before the period ends so students have time to return all materials. They save their worksheets.

This entire procedure is repeated the following day. Some time will probably be available at the end of the next day to summarize, share results, and give feedback.

Discussion

The results of the different experiments can be discussed with the whole class. Particular attention should be paid to using the materials to check table entries. Two of the master wizard experiments (BEANS AND WAYS and STAMPING RODS) are quite difficult in that they involve complicated rules. The discussion of the two "Group of Four" worksheets should follow from the presentation of student's work. Have a group present their "best" conditional machine for the whole class to attempt. Have them present enough of the table for the other groups to work on. Do this several times. It is not necessary to have every group present at this time. Have the groups describe how they "got it". You could also at this point write EASY SPEAK descriptions of those rules presented either as a whole class or in small groups. Follow a similar procedure for interesting solved examples from the WIZARD program. If you don't insist that they use solved examples, the class may be involved in mathematics above their level.

THE HUMAN COMPUTER

Format
Whole class and groups of four

Time
Two class periods

Materials Needed
PET computers
EASY SPEAK program
EASY SPEAK summary
HUMAN COMPUTER worksheet

Background

Having been introduced to EASY SPEAK statements in the MYSTERY MACHINES activity, students will now create their own mystery machines. This HUMAN COMPUTER activity provides students with the experience of writing conditional statements in EASY SPEAK. Students will need to write EASY SPEAK statements in order to create machines on the concluding computer activity for this unit. Writing conditional statements is also needed for other units in this project.

Purpose

- To practice writing conditional statements in the syntax of the EASY SPEAK language.
- To evaluate other students' conditional statements in EASY SPEAK.
- To introduce the EASY SPEAK summary page and the EASY SPEAK program.

Teacher Preparation

Review the EASY SPEAK statements used so far. Load the EASY SPEAK program on both computers.

Run off copies of the HUMAN COMPUTER worksheet and the EASY SPEAK summary page for each student.

Load EASY SPEAK program into both computers.

Activity

Review

Divide the class into groups of four. Have students take out the EXAMPLES OF CONDITIONALS worksheet from their folders. Tell them that the exact form in which the conditional statements are written is

important in order to have the computer understand. As a check, you may have students write the conditional statements for the unmatched table (table D), or you can write the description of any other tables as a group.

Explain that the computer understands EASY SPEAK but that each statement must be very carefully written. Two rules that have been assumed up to now are important to point out:

1. The machine must know what to do with ANY output.
2. The machine must have only ONE output for a given input.

Write these two rules on the board and explain to the class that the intent of the two rules is that there will be exactly ONE input for ANY input. Examples of conditions that do not satisfy these two rules would help demonstrate their meaning. Picking an input for which the output is ambiguous will demonstrate to the class why these examples violate a rule.

Examples of Improper Machines:

```
1.CONDITION:  INPUT<10
  OUTPUT= INPUT + 2

2.CONDITION:  INPUT>5
  OUTPUT= INPUT + 6
```

or

```
1.CONDITION:  FACT(3,INPUT)
  OUTPUT= INPUT + 1

2.CONDITION:  FACT(6,INPUT)
  OUTPUT= Input + 4
```

or

```
1.CONDITION:  INPUT>9
  OUTPUT= 8

2.CONDITION:  EVEN(INPUT)
  OUTPUT= INPUT
```

Group of Four Activity

Each student individually completes the HUMAN COMPUTER worksheet. Questions one and two can be checked by the group or corrected later by the teacher. Question three is to be used by the human computer. Each student will have a turn to be the human computer and will give output requested by the other three. Their task will be to discover what EASY SPEAK statements the computer is using. When they think they have the same statements, they should write them down and then check them with the human computer. If they are not exactly the same, the whole group should check to see if the new statements do exactly the same thing. If they did not correctly solve the human computer's problem, mark it with a star. These starred problems may be looked at separately or used in the closing discussion.

Hand out copies of the EASY SPEAK summary and have students go through with you marking those items that have been used so far. Tell them that the EASY SPEAK program can use all of them, but they must be written carefully for the computer to understand.

Demonstrate Easy Speak Program

The last ten minutes of class, or during the next class, write a simple EASY SPEAK machine on the chalkboard. Have the class gather around the computer (a student may be designated to operate the second computer). Demonstrate the EASY SPEAK program options to the class. Note the choices available. Select "CREATE A MACHINE" and type in the machine from the chalkboard. As you type in the CONDITION and then the OUTPUT, point out the blinking "T" which appears after you press RETURN. This indicates the machine is thinking about your statement. Tell the class there is a maximum of five statements allowed. When you have completed your machine, you should return to the menu. Select "RUN THE MACHINE" and try several INPUTS. If your machine is not carefully constructed, the computer will tell you when it comes upon a problem. If a problem exists in your program, you will need to return to the MENU and select the "EDIT THE MACHINE" option and correct the error. You can change the statement the cursor is indicating or erase the statement completely with "Shift@".

Discussion

As time permits, probably the second day of this activity, discuss the human computer activity. You may correct the worksheet at this time. A good discussion point would follow from the human computer machines that were not correctly solved (they were marked with a star). This is also a chance to explore the use of other EASY SPEAK statements from the SUMMARY SHEET.

EASY SPEAK

Format
Whole class and individual group computer time

Time
half a period as a group and several days of individual computer time

Materials Needed
PET computers
EASY SPEAK program
EASY SPEAK worksheet

Background

This is the concluding activity for this unit. The use of the EASY SPEAK program and the EASY SPEAK SUMMARY will permit students working in pairs to create mystery machines as well as having them try mystery machines created by students in other classes.

Purpose

- To have students create their own conditinal INPUT-OUTPUT machines using the EASY SPEAK program.
- To check students' understanding of the syntax of the EASY SPEAK language.
- To have students save and retrieve EASY SPEAK programs through the NETWORK.

Teacher Preparation

Run off copies of EASY SPEAK WORKSHEET for each student.
Load the EASY SPEAK program into both computers. Prepare four problems to present to the class as a review. Two of these should be EASY SPEAK statements; students will construct a table. The other two should be tables; students will write the EASY SPEAK statements. Preselect a machine from the Network if you are going to demonstrate NETWORK EASY SPEAK.

EXAMPLES

1. CONDITION: FACT(6,INPUT)
   OUTPUT = 100-INPUT
2. CONDITION: NOT(FACT(6,INPUT))
   OUTPUT = INPUT+7

1. CONDITION: (EVEN(INPUT)) AND (INPUT<10)
   OUTPUT = 2*INPUT
2. CONDITION: (ODD(INPUT)) AND (INPUT<10)
   OUTPUT = 3*INPUT
3. CONDITION: INPUT>=10
   OUTPUT = 10

Examples of tables you could use are in the margin.

INPUT	OUTPUT
3	2
4	3
5	4
6	19
10	31
11	34
12	36
25	76

INPUT	OUTPUT
8	65
6	49
7	56
5	40
3	24
9	72
1	8
2	17
4	33

Set up a schedule for computer use that will start halfway through this period. Pairs of students will need to work for about 20 minutes on the computer using the EASY SPEAK program.

NOTE: If students will be saving or retrieving programs, they will need to work on the computer that is connected to the NETWORK.

Prepare material from the regular curriculum for the class to work on after completing the EASY SPEAK WORKSHEET.

Activities

Review

Begin by asking the class to recall as many items from the EASY SPEAK SUMMARY as they can. Write the list on the blackboard.

Present your first prepared EASY SPEAK statements and ask the class to give you the output for input that you select. Maximize involvement by asking several people about the same input.

The second example is not difficult, but the three conditions should be checked with the class to show that there is exactly one output for any input. If you plan to demonstrate the NETWORK retrieval of an EASY SPEAK program, select one before class that you feel the class can solve. Some NETWORK programs are complex and should be avoided as first examples.

Present your prepared tables. Students may want to suggest more inputs to help them determine what the EASY SPEAK statements might be. Have several students write the EASY SPEAK statements on the board. Ask the other students to check the table against the statements. Seek out different statements where possible.

Using Easy Speak

Type in one student's statements to demonstrate how to begin the EASY SPEAK program. After typing it in, return to the menu and select "RUN THE MACHINE". (The Demonstrate Easy Speak Program section of the previous

activity describes some details about running and editing your machines.) Ask about some larger numbers. There is a five statement limit for EASY SPEAK programs. This will generally not be a problem for students' work.

Network Easy Speak

As a further example, use the computer connected to the NETWORK to retrieve your pre-selected machine. Explain to the class that other students have saved their EASY SPEAK machines and that you will now pick one for them to try.

select "LOAD A MACHINE" and follow the procedures to enter the machine you have pre-selected from the NETWORK. Select "RUN THE MACHINE" and have students as a whole or in small groups work out the selected machine. Let them suggest inputs and have them write out EASY SPEAK statements when they think they have it. You could then select either "TEST YOURSELF ON THE MACHINE" or "EDIT THE MACHINE" to check their results.

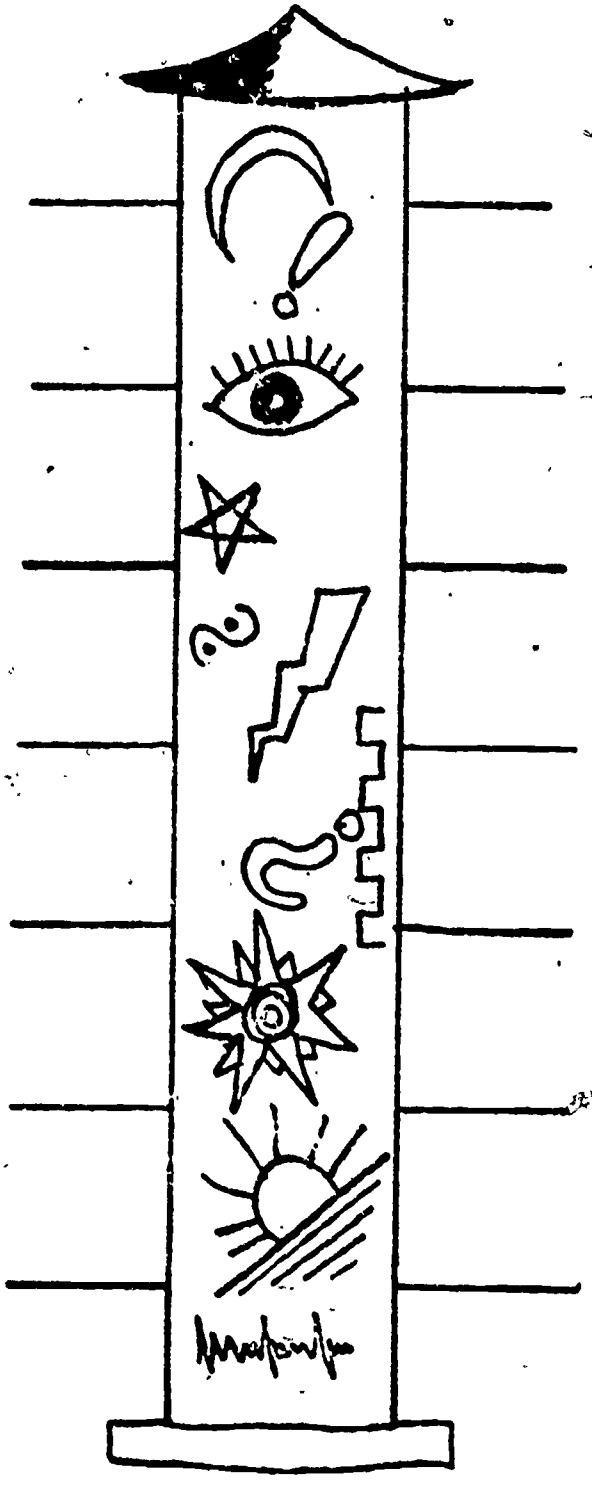
> NOTE: A list of "Network Programs to Try" could be posted as a starter for students. Students may add to the list if they write or find one they think is interesting.

Set the schedule for computer use and have the rest of the class begin the EASY SPEAK worksheet.

Discussion

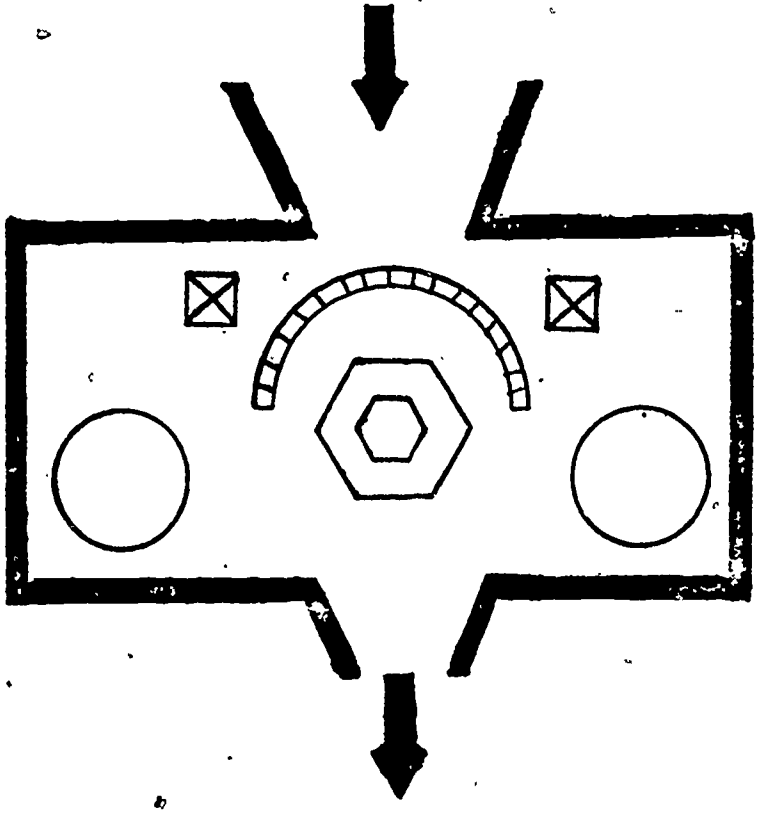
After each pair of students has had a chance to work through the EASY SPEAK program to write programs of their own, take some time to have students present examples of their work from the worksheet, or examples they select from the NETWORK. The discussion might now extend to the possible uses of additional EASY SPEAK statements, which can be found on the EASY SPEAK SUMMARY sheet.

Appendix



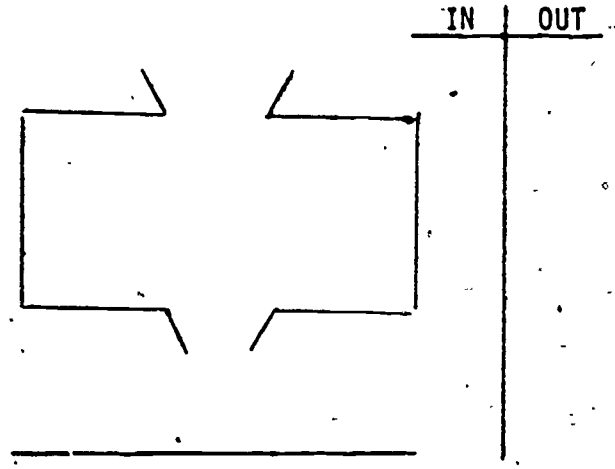

MYSTERY
TOWER

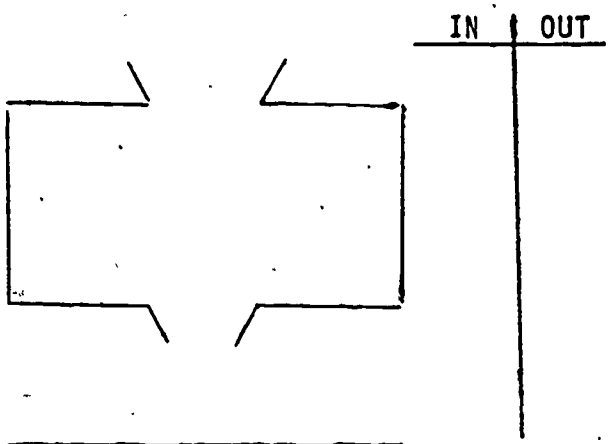
INPUT-OUTPUT MACHINE

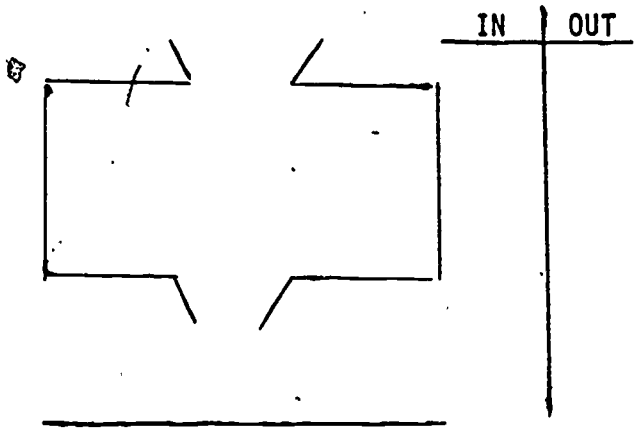


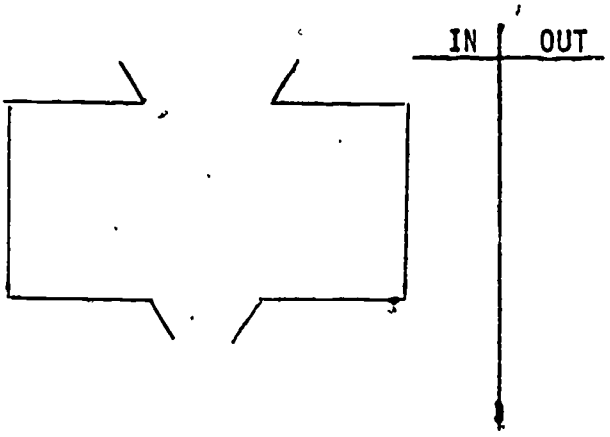
INPUT	OUT PUT

OTHER MACHINES









INPUT - OUTPUT DISCUSSION WORKSHEET

Do these examples:

A

INPUT	OUTPUT
1	2
2	5
3	8
4	11
5	14
6	17
7	20

The rule is:

_____

_____

B

INPUT	OUTPUT
4	11
6	15
2	7
1	5
3	9
5	13
10	23

The rule is:

_____

_____

C

INPUT	OUTPUT
3	13
2	9
6	25
8	33
10	41
1	5
4	17

The rule is:

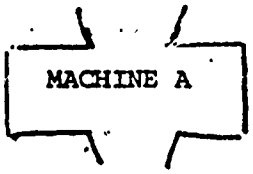
_____

_____

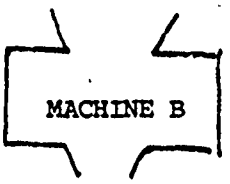
Make a list of hints you can give someone (without telling the answer) to help them discover these rules.

List other hints you know about.

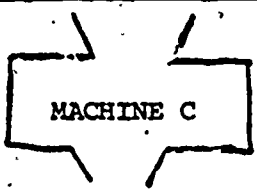
MYSTERY MACHINES



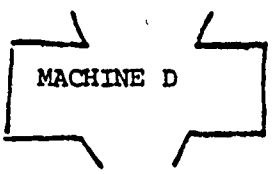
IN	OUT
1	7
2	9
3	11
4	13
5	15
6	17



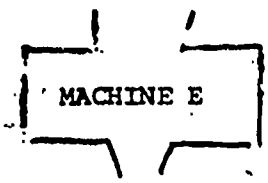
IN	OUT
1	2
2	22
3	6
4	44
5	10
6	66



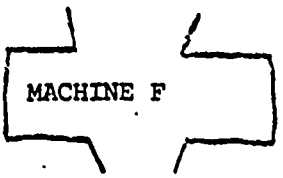
IN	OUT
1	2
2	4
3	6
4	24
5	30
6	36



IN	OUT
7	5
8	5
9	5
10	13
11	13
12	13



IN	OUT
5	15
6	14
7	13
8	12
9	11
10	10



IN	OUT
3	8
4	11
5	14
6	17
7	20
8	23

EXAMPLES OF CONDITIONALS

Listed below are six sets of conditional statements in EASY SPEAK and six tables. In the spaces below, write the letter of the table that matches the numbered statements. One of the six does not match.

I._____ III._____ V._____

II._____ IV._____ VI._____

I. 1. CONDITION: INPUT > 6
OUTPUT= 20-INPUT.

2. CONDITION: INPUT ≤ 6
OUTPUT= 2*INPUT-1

II. 1. CONDITION: EVEN(INPUT)
OUTPUT= 8

2. CONDITION: ODD(INPUT)
OUTPUT= INPUT+5

III. 1. CONDITION: FACT(3,INPUT)
OUTPUT= INPUT*INPUT

2. CONDITION: NOT FACT(3, INPUT)
OUTPUT= 3*INPUT

IV. 1. CONDITION: EVEN(INPUT)
OUTPUT= INPUT-1

2. CONDITION: ODD(INPUT)
OUTPUT= 13-INPUT

V. 1. CONDITION: INPUT < 6
OUTPUT= 3*INPUT

2. CONDITION: INPUT ≥ 6
OUTPUT= INPUT*INPUT

VI. 1. CONDITION: FACT(5,INPUT)
OUTPUT= 2*INPUT+2

2. CONDITION: NOT(FACT(5,INPUT))
OUTPUT= INPUT+7

A	
INPUT	OUTPUT
3	5
4	7
5	9
6	11
7	13
8	12
9	11
10	10
11	9

B	
INPUT	OUTPUT
1	3
2	6
3	9
4	12
5	15
6	36
7	21
8	24
9	81

C	
INPUT	OUTPUT
2	8
3	8
4	8
5	10
6	8
7	12
8	8
9	14
10	8

D	
INPUT	OUTPUT
4	8
5	10
6	12
7	14
8	16
9	13
10	14
11	15
12	16

E	
INPUT	OUTPUT
2	1
6	5
1	12
4	3
5	8
9	4
3	10
8	7
7	6

F	
INPUT	OUTPUT
1	8
2	9
3	10
4	11
5	12
6	13
7	14
8	15
9	16

WORKSHEET A

Complete the table for each of the following EASY SPEAK programs.

A. 1. CONDITION: ALWAYS
OUTPUT= 3*INPUT+5

INPUT	OUTPUT
1	
2	
3	
4	
5	
6	
7	
8	

B. 1. CONDITION: EVEN(OUTPUT)
OUTPUT= 2*INPUT-3

2. CONDITION: ODD(INPUT)
OUTPUT= INPUT+13

INPUT	OUTPUT
6	
7	
8	
9	
10	
11	
12	
13	

C. 1. CONDITION: INPUT < 6
OUTPUT= 30-INPUT

2. CONDITION: INPUT > 6
OUTPUT= INPUT-4

3. CONDITION: INPUT=6
OUTPUT= INPUT*INPUT

INPUT	OUTPUT
8	
3	
6	
5	
9	
2	
15	

D. 1. CONDITION: FACT(3,INPUT)
OUTPUT= 1

2. CONDITION: NOT(FACT(3,INPUT))
OUTPUT= 5*INPUT+1

INPUT	OUTPUT
1	
2	
3	
4	
5	
6	
7	
8	

WORKSHEET B

Write an EASY SPEAK program for each table.

(HINT: You can use < , <= , > , >= , ODD, EVEN, FACT, NOT, +, * , -)

A.

INPUT	OUTPUT
4	6
5	8
6	10
7	12
8	17
9	16
10	15
20	5

1. CONDITION:
   OUTPUT=
2. CONDITION:
   OUTPUT=

B.

INPUT	OUTPUT
5	21
6	26
7	31
8	36
11	30
12	33
13	36
25	72

1. CONDITION:
   OUTPUT=
2. CONDITION:
   OUTPUT=

C.

INPUT	OUTPUT
1	29
2	28
5	15
6	24
7	23
12	18
20	60
24	6
25	75

1. CONDITION:
   OUTPUT=
2. CONDITION:
   OUTPUT=

D.

INPUT	OUTPUT
11	11
7	11
13	11
4	10
10	28
3	11
2	4
8	22
6	16

1. CONDITION:
   OUTPUT=
2. CONDITION:
   OUTPUT=

WIZARD'S WORKSHOP

EXPERIMENT TITLE

TABLE

DESCRIPTION OF TABLE

*CRITERIA QUESTION
FOR__________YOU GET__________

EXPERIMENT TITLE

TABLE

DESCRIPTION OF TABLE

*CRITERIA QUESTION
FOR__________YOU GET__________

EXPERIMENT TITLE

TABLE

DESCRIPTION OF TABLE

*CRITERIA QUESTION
FOR__________YOU GET__________

EXPERIMENT TITLE

TABLE

DESCRIPTION OF TABLE

*CRITERIA QUESTION
FOR__________YOU GET__________

WIZARD'S WORKSHOP

CUBE TOWER PAINTING

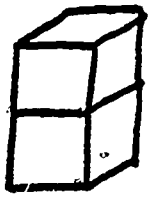
MATERIALS:
10 or 12 cubes

With the first tower there are 5 squares to paint.

(4 sides and the top)

DON'T COUNT THE BOTTOM SQUARE

With the second tower there are 9 squares to paint.



How many squares for the 3rd tower? 4th? Make a table.

	SQUARES
1	5
2	9
3	

*Suppose you built the 99th tower.
Suppose you had to paint each square on the tower.
How many squares would you have to paint?

FOR MASTER WIZARDS

The first tower is the same, but for each succeeding tower you add two cubes to be painted.

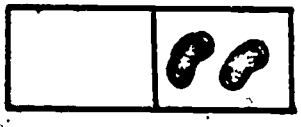
WIZARD'S WORKSHOP

BEANS AND WAYS

MATERIALS:
10 or 12 beans
(or counters)
2 cups or squares
drawn on paper

With just 1 bean, there are 2 ways to put it into the containers.

With 2 beans, you can put them into the containers 3 different ways.



How many ways for 3 beans? 4? Make a table.

BEANS	WAYS
1	2
2	3
3	

*Suppose you had 50 beans.
How many ways could you put the 50 beans in the containers?

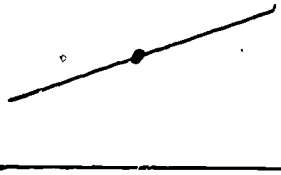
FOR MASTER WIZARDS

All is the same, except that there are three containers.

WIZARD'S WORKSHOP

SECTIONS ON A LINE

With one point, you would have 2 sections



With 2 points, there would be 3 sections.

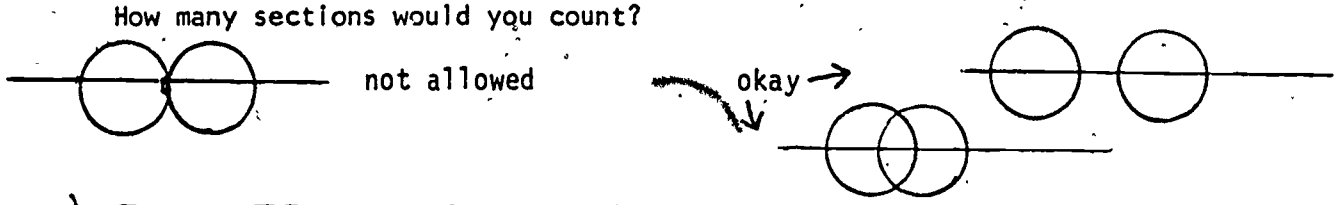
Continue making points and counting sections.

POINTS	SECTIONS
1	2
2	3
3	

*If you put 20 points on a line segment, how many sections would you count?

FOR MASTER WIZARDS

Instead of points, you draw circles (the sides don't touch).
How many sections would you count?



WIZARD'S WORKSHOP

STAMPING RODS

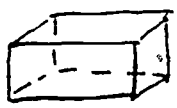
MATERIALS
1 rod of each color

Suppose the white Cuisenaire rod were a rubber stamp. It would stamp a shape like this: ◘ How many stamps would it take to cover each of the other rods?

The white rod takes 6 stamps.

(Don't forget the bottom!)

The red rod takes 10 stamps.



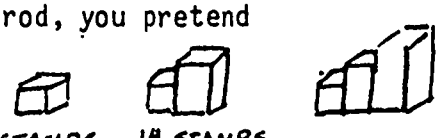
CONTINUE THE STAMPING.

ROD	STAMPS
1-white	6
2-red	10
3-light gr.	
4-purple	
5-yellow	
6-dark gr.	
7-black	
8-brown	
9-blue	
10-orange	

*Suppose you had a rod 32 units long; how many stamps?

FOR MASTER WIZARDS:

Instead of starting with the next rod, you pretend to glue it to the previous rod (s):



...etc.

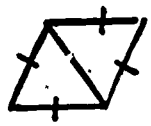
1st 2nd 3rd

WIZARD'S WORKSHOP

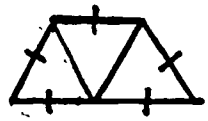
A ROW OF TRIANGLES

WITH 1 TRIANGLE, THE PERIMETER IS 3 UNITS.

WITH 2 TRIANGLES, THE PERIMETER IS 4 UNITS.



WITH 3 TRIANGLES, THE PERIMETER IS 5 UNITS.



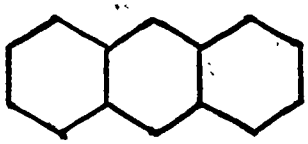
CONTINUE THE EXPERIMENT.

Tri-angles	peri-meter
1	3
2	4
3	
4	

*If you lined up 100 equilateral triangles in a row what would the perimeter measure?

FOR MASTER WIZARDS:

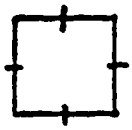
Instead of triangles, use regular hexagons.



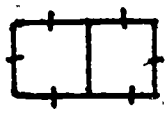
WIZARD'S WORKSHOP

A ROW OF SQUARES

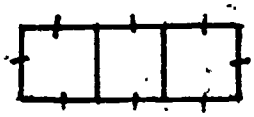
WITH 1 SQUARE, THE PERIMETER IS 4 UNITS.



WITH 2 SQUARES, THE PERIMETER IS 6 UNITS.



WITH 3 SQUARES, THE PERIMETER IS 8 UNITS.



CONTINUE THE EXPERIMENT.

Squares	Perimeter
1	4
2	6
3	
4	

*If you lined up 100 squares in a row (), what would the perimeter measure?

FOR MASTER WIZARDS:

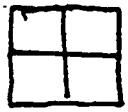
Instead of squares, use pentagons.

WIZARD'S WORKSHOP

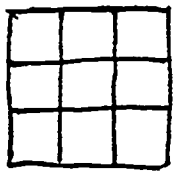
SQUARES FROM SQUARES

For a square with side 1, the perimeter is 4.

For a side of 2, the perimeter is 8.



For a side of 3, it's 12.



Continue the experiment.

LENGTH OF SIDE	PERIMETER
1	4
2	8
3	
4	

*FOR A SQUARE OF SIDE 47, WHAT IS THE PERIMETER?

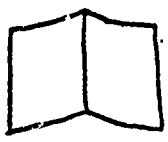
FOR MASTER WIZARDS

INSTEAD OF FINDING THE PERIMETER FOR EACH SIZE SQUARE, FIND THE AREA.

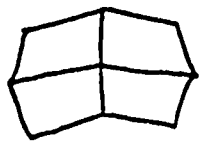
WIZARD'S WORKSHOP

PAPER FOLDING

With 1 fold, you would have 2 sections.



With 2 folds, you'd have 4 sections.



(REFOLD EACH TIME)

Make a table. Continue the folding.

FOLDS	SECTIONS
1	2
2	4

*IF IT WERE POSSIBLE TO FOLD THE PAPER 11 TIMES, HOW MANY SECTIONS WOULD YOU HAVE?

EASY SPEAK SUMMARY

+	addition	=	equals
-	subtraction	<	less than
*	multiplication	>	greater than.
/	division.	<=	less than or equal to
↑	exponentiation	>=	greater than or equal to
		<>	not equal to

ODD ODD (INPUT)

EVEN EVEN (INPUT)

ALWAYS Is always true.

NEVER Is never true.

FACT FACT(5,INPUT)
Will be true if 5 is a factor of the input.

NOT NOT(INPUT < 10)
Change the truth value of the statement.

AND (5,INPUT) AND (EVEN(INPUT))

OR (EVEN(INPUT)) OR (5 < INPUT)

SQR SQR(9) SQR(INPUT)
SQR(9) gives 3, the square root of 9.

INT INT(4.2) INT(INPUT/5)
INT(4.2) gives the value 4

RMD RMD(8,3)
RMD(8,3) gives 2, the remainder of 8÷3.

MAXIMUM MAXIMUM(INPUT,10)

MINIMUM MINIMUM(INPUT*3,20)

LCM LCM(4,6) LCM(INPUT,24)
LCM(4,6) gives 12, the least common multiple of 4 and 6.

GCD GCD(INPUT,5) GCD(20,8)
GCD(20,8) gives 4, the greatest common divisor of 20 and 8.

RANDOM RANDOM(2,10) RANDOM(INPUT,20)
RANDOM(2,10) will randomly pick either
2, 3, 4, 5, 6, 7, 8, 9, or 10.

HUMAN COMPUTER WORKSHEET

1. COMPLETE THE TABLE.

   1. CONDITION: FACT(4, INPUT)

      OUTPUT = INPUT - 3

   2. CONDITION: NOT(FACT(4, INPUT))

      OUTPUT = 25 - INPUT

INPUT	OUTPUT
3	
4	
7	
12	
6	
8	
20	
9	
16	

2. WRITE THE EASY SPEAK STATEMENTS FOR THE GIVEN TABLE.

   1. CONDITION:

      OUTPUT =

   2. CONDITION:

      OUTPUT =

INPUT	OUTPUT
3	7
4	6
5	11
6	8
7	15
8	10
9	19
10	12
11	23

3. WRITE EASY SPEAK STATEMENTS FOR A MACHINE FOR YOUR GROUP TO TRY AND DISCOVER. DO NOT USE ANY NUMBERS LARGER THAN 10 IN YOUR STATEMENTS.

   1. CONDITION:

      OUTPUT =

   2. CONDITION:

      OUTPUT =

EASY SPEAK WORKSHEET

Write EASY SPEAK statements of your own that use the items given. (You may use additional items.) Write your statements carefully and neatly, using the correct EASY SPEAK form.

I. This machine should use:
+, EVEN, 8, =

1. CONDITION: _____
OUTPUT= _____

2. CONDITION: _____
OUTPUT= _____

Write the table, too.

INPUT	OUTPUT
4	
5	
6	
7	
8	
20	
21	

II. This machine should use:
FACTOR, 20-INPUT, 11, *

1. CONDITION: _____
OUTPUT= _____

2. CONDITION: _____
OUTPUT= _____

Write the table.

INPUT	OUTPUT
3	
4	
5	
6	
7	
8	
9	
20	

III. This machine should use:
21, <=, ↑, -, ÷

1. CONDITION: _____
OUTPUT= _____

2. CONDITION: _____
OUTPUT= _____

Write the table.

INPUT	OUTPUT
2	
3	
4	
5	
30	
40	
21	

IV. This machine should use:
>, INPUT-2, *, 12

1. CONDITION: _____
OUTPUT= _____

2. CONDITION: _____
OUTPUT= _____

Write the table.

INPUT	OUTPUT
10	
20	
30	
40	
18	
24	
100	

WIZARD 1.3

READY.

```
1000 REM---------MAIN------------
1010 GOSUB 1400,INIT :GOSUB6000,OPENING GRAPHICS:IF FNQ(T)THEN1190 END
1020 GOSUB2000,PREREQ. WARNING
1030 GOSUB2200,GET NAMES:GOSUB2400,RULES
1040 FOR JRNY=1 TO 20
1050 :GOSUB2600,PICK NAME:GOSUB2800,GET SEED/MACH.
1060 :IF ESC THEN1180
1070 :PRINT"⌧":POKE CR,12:X=10:Y=10:GOSUB8800:PRINT"AWAY WE GO!"
1080 :AS=8:GOSUB9000:PRINT"⌧";:GOSUB6200,PAINT MACH
1090 :FOR TABLE=1 TO 35
1100 ::GOSUB3000,INPUT,GUESS,QUIT
1110 ::ON TYPE GOTO 1120,1130,1150
1120 :::GOSUB3200:GOTO 1160
1130 :::GOSUB3400:IF FF=1THENFF=0:GOTO1150
1140 :::GOSUB3800:GOTO 1160
1150 :::TA=35:NEXTTA:GOTO1170
1160 :NEXT TABLE:OW=1:GOTO1090
1170 OW=0:NEXT JRNY
1180 GOSUB3600 EVALUATION:POKE144,P:END
1190 PRINT"⌧":POKE CR,12:PRINT"GOODBYE!":POKE144,P:END
1200 REM**PLACE DATA
1210 DATA PEACEFUL,TRANQUIL,CALM,THREATENING,CREEPY
1220 DATA HAZARDOUS,DANGEROUS,TREACHEROUS,PERILOUS,DEADLY
1230 DATA HAUNTED,ENCHANTED,MAGIC,GLOWING,FROZEN
1240 DATA HUGE,GOLDEN,MYSTIC,MYTHICAL,MYSTERIOUS
1250 DATA CAVE,STRING OF PEARLS,BEE HIVE,BOX OF BAND AIDS,DESERT,GOLD NUGGET
1260 DATA GROTTO,TREASURE CHEST,MARSH,HUGE DIAMOND,CASTLE,"RUBY RING "
1270 DATA LIGHTHOUSE,MAGIC LAMP,SWAMP,GOLD CHAIN,GRAVEYARD,CRYSTAL BALL
1280 DATA GALLOWS,TIME PORTAL
1400 REM**INIT
1410 PRINT"⌧"CHR$(142):POKE59468,14
1420 DI  NAME$(6),FUN$(21),EARN$(20),INPT(35),SEED(20)
1430 D$="▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒":DEFFNQ(T)=PEEK(151)=3ORPEEK(151)=4
1440 DEF FNES(C)=(((PEEK(151)=15ORPEEK(151)=64)ANDPEEK(152)=1)ORFNQ(T))
1450 GOSUB1600:Z=RND(-RND(0)):R$=CHR$(13):CR=59468:S=255:CX=1:SND=1
1460 P=PEEK(59271):P=(-88*(P=170ORP=29)-49*(P=210)):POKE144,P:P=P-3
1470 RETURN
1600 REM**SETUP FUNCTIONS
1610 DATA A11,A55,B14,C45
1620 DATA A22,B15,C14,D41X,D51X
1630 DATA A33,A66,B25,C52X,C43X,D52X,D42X
1640 DATA A77,B46X,C56X,D53X,D43X
1650 FORI=0TO39:READA$:NEXT
1660 FORI=0TO20:READFUN$(I):NEXT
1670 RETURN
2000 REM**PREREQUISITE WARNING
2010 POKE CR,14:PRINT"⌧♠|-!!.........▒▒▒▒":IF FNES(C)THEN2080
2020 PRINT"♠_─ Γ, ─┐♠─ ─Γ─ | |♠??!!
2030 PRINT"▒▒▒▒F YOU HAVE NOT SEEN THE ▒ ▶|─ | |Γo─
2040 PRINT"▒▒o♠♠_─ WILL OVERPOWER YOU....
2050 PRINT"▒▒F YOU HAVE NOT SEEN THE ▒ ▶|─ | ♠─ |/▀";
2060 PRINT"▒▒o♠♠_─ WILL BOGGLE YOUR MIND!!!....
2070 AS=2:GOSUB9000:X=0:Y=22:GOSUB8800:GOSUB8600:IF ESC=1THEN2120
2080 POKE CR,12
```

```
2090 PRINT"▯▯▯▯▯▯▯▯▯▯▯YOU ARE ENTERING THE MAGIC WORLD!■"
2100 FOR I=1TO500:NEXT:AS=3:GOSUB9000:X=0:Y=21:GOSUB8800:IF FNQ(T)THEN1190 END
2110 GOSUB8600
2120 RETURN
2200 REM**GET NAMES
2210 POKE CR,12:PRINT"▯▯▯▯▯▯":IFFNES(C)THENNA$(1)=" 1":GOTO2390
2220 PRINT"TYPE IN THE NAME (AND PRESS ▯RETURN■)"R$"▯FOR EACH PERSON IN YOUR";
2230 PRINT" GROUP."R$"▯PRESS ▯SHIFT &■ TO GO ON.▯▯▯":AS=4:GOSUB9000:FORI=1TO6
2240 :PRINT"NAME"I"■: ";:B$="":GOSUB8200:IFLEN(B$)=0 ORB$="▒"ORB$="-"THEN2270
2250 :IFLEN(B$)=0 ORB$="▒"ORB$="-"THEN2270
2260 :NAME$(I)=B$:NEXT I:NBIG=6:GOTO2390
2270 NBIG=I:IFB$="-"THENNAME$(I)=STR$(I):GOTO2390
2280 IF I>3THEN2390
2290 PRINT"▯":POKE CR,14:ON I GOTO2210,2300,2360
2300 NAME$=CHR$(128 OR ASC(NA$(1)))+RIGHT$(NA$(1),LEN(NA$(1))-1)
2310 PRINT"▯":X=0:Y=10:GOSUB8800
2320 PRINT" ▯OU ARE VERY BRAVE, ";NAME$;","R$"▯ TO TRY THIS ALONE!
2330 PRINT"▯ ▯OU HAVE ONE MORE CHANCE.▯"R$"-O YOU WANT TO GET HELP ";
2340 AS=10:GOSUB9000:GOSUB7800:IF YES=1 THEN PRINT"▯▯▯▯":POKE CR,12:GOTO2240
2350 GOTO2390
2360 PRINT"▯▯▯▯▯▯NLY TWO TRAVELERS?  |HERE'S SAFETY IN▯"R$"NUMBERS, BUT YOU";
2370 PRINT" HAVE DECIDED TO TAKE▯"R$"A CHANCE."R$"▯▯▯....PROCEED....▯▯▯▯
2380 AS=1:GOSUB9000:GOSUB8600
2390 RETURN
2400 REM**RULES
2410 POKE CR,14:IF FNES(C)THEN2550
2420 PRINT"▯▯▯▯▯-,♠♣_- IS A JOURNEY."
2430 PRINT"▯    -ACH PLACE YOU COME TO WILL PRESENT"
2440 PRINT"YOU WITH A MYSTERY MACHINE.     ▯OU MAY"
2450 PRINT"INPUT AS MANY NUMBERS AS YOU LIKE."
2460 PRINT"▯    OHEN YOU ARE READY TO GUESS, THE"
2470 PRINT"MACHINE WILL ASK YOU FOUR QUESTIONS."
2480 PRINT"▯    ,F YOU GET THEM ALL RIGHT YOU WILL"
2490 PRINT"WIN A TREASURE."
2500 PRINT"▯    ,F YOU GUESS AND MISS YOU WILL LOSE"
2510 PRINT"A TREASURE."
2520 PRINT"▯     ▯OU MAY AT ANY TIME LEAVE THE PLACE"
2530 PRINT"YOU'RE IN AND GO ON TO ANOTHER.":AS=6:GOSUB9000
2540 X=0:Y=24:GOSUB8800:GOSUB8600
2550 RETURN
2600 REM**PICK A NAME
2610 NAME=NAME+1:IF NAME=>NBIG THEN NAME=1
2620 NAME$=NAME$(NAME)
2630 RETURN
2800 REM**GET SEED/MACHINE
2810 PRINT"▯":POKE CR,14:PRINT"▯▯▯-, ";:ESC=0
2820 PRINT CHR$(128 OR ASC(NA$(NA)))+RIGHT$(NA$(NA),LEN(NA$(NA))-1)
2830 PRINT"▯ \OURNEY ONWARD!!
2840 PRINT"▯    ▯OU WILL BE TRANSPORTED TO A MYSTERY▯ PLACE!"
2850 IFJR>1THENPRINT"▯(TO QUIT AND SEE YOUR SCORE, ENTER ▯■■)"
2860 PRINT"▯    -HOOSE A LEVEL OF DIFFICULTY FROM▯    0 (EASY) TO 9 (HARD!) ";
2870 AS=1:GOSUB9000:L=0:GOSUB7600:IFNM$="Q"THENESC=1:GOTO2970
2880 PRINT"▯/OW PICK A LOCATION FROM 1 TO 100 ";:W1=INT(NM+.5):IFW1>9THENW1=9
2890 AS=7:GOSUB9000:L=2:GOSUB7600:IFNM$="Q"THENESC=1:GOTO2970
2900 NM=NM+(NM=>1):W2=INT(NM/10):IFW2>9THENW2=9
2910 W3=INT(NM-W2*10):IFW3>9THENW3=9
2920 SEED=100*W1+10*W2+W3+1
2930 MACH=1-(SEED>399)-(SEED>699)-(SEED>899)
2940 RESTORE:FORI=0TO W1:READ PLACE$:NEXT
2950 RESTORE:FORI=0TO9:READA$:NEXT:FORI=0TOW2:READW2$:NEXT:PL$(1)=PL$+" "+W2$
```

```
2960 RESTORE:FORI=0TO19:READA$:NEXT:FORI=0TOW3:READPLACE$(2),TREASURE$:NEXT
2970 RETURN
3000 REM**GET COMMAND
3010 X=0:Y=20:GOSUB8800:POKE CR,12:POKE158,0
3020 PRINTNAME$", PLEASE GIVE A  "R$"NUMBER TO INPUT              "
3030 PRINT"TYPE G TO GUESS"R$"TYPE Q TO QUIT THIS ONE";
3040 X=16:Y=21:GOSUB8800:B$="":TYPE=1
3050 GOSUB8000:A=ASC(A$):IF LEN(B$)>2THEN3100
3060 IFA$="-"THENTYPE=3:RETURN
3070 IF((A>47 AND A<58)OR A=45)AND TYPE=1THEN3130
3080 IF((AAND127)=ASC("G"))AND(LEN(B$)=0)THEN TYPE=2:GOTO3130
3090 IF((AAND127)=ASC("Q"))AND(LEN(B$)=0)THEN TYPE=3:GOTO3130
3100 IF(A=20)ANDLEN(B$)THENPRINT"  ";:B$=LEFT$(B$,LEN(B$)-1):TYPE=1:GOTO3050
3110 IFA=13 ANDLEN(B$)THEN3140
3120 GOTO3050
3130 B$=B$+A$:PRINTA$;:IFLEN(B$)<8THEN3050
3140 AS=5:GOSUB9000:RETURN
3200 REM**INPUT NUMBER
3210 IFTABLE>(W1+6)THENTC=TC+1
3220 IFTABLE<19ANDOW=0THENGOSUB7000:GOSUB7000
3230 INPT(TABLE)=VAL(B$):X=23:Y=0:GOSUB8800
3240 IFTABLE=1 THEN PRINT" INPUT| OUTPUT "
3250 X1=24:Y1=0:IF TABLE>18 THEN X1=3:Y1=-18
3260 IF TABLE<>19 OR OW THEN3350
3270 PRINT"   INPUT| OUTPUT   "
3280 PRINT "   ------+-------  ";
3290 PRINT"     INPUT| OUTPUT "
3300 FOR T1=1 TO 18
3310 :X=24:Y=T1+1:GOSUB8800:PRINTINPT(T1);:X=29:GOSUB8800:PRINT"|";
3320 :INPT=INPT(T1):GOSUB4000 CALC.
3330 :PRINT OTPT;
3340 NEXT T1
3350 X=X1:Y=TABLE+Y1+1:GOSUB8800
3360 PRINT"         "INPT(TABLE);:X=X1+5:GOSUB8800:PRINT" ";
3370 INPT=INPT(TABLE):GOSUB4000
3380 PRINT OTPT;:RETURN
3400 REM**GUESS
3410 GOSUB7400 NO TREASURE
3420 PRINT"";:GOSUB1-1000
3430 Q(1)=-1:Q(2)=-1:Q(3)=-1
3440 FOR Q=1 TO 4
3450 :Z=RND(-RND(0)):REM RANDOM SEED
3460 :INPT=INT(RND(Z)*10):IFMA>1THENINPT=INT(INPT*2.5*MA)
3470 :IF F3=67 AND(Q(1)>K OR Q(2)>K OR IN>K)THEN IN=INT(RND(-RND(0))*K+1)
3480 :IF F3=67ANDQ=4 THEN IN=(IN+2)*K
3490 :IF IN=Q(1)OR IN=Q(2)OR IN=Q(3)THEN 3450
3500 :Q(Q)=INPT:GOSUB4000:X=10:Y=10+Q:GOSUB8800:PRINTINPT
3510 :X=X+8:GOSUB8800:PRINT"|     "";:L=4:GOSUB7600:IFNM$="Q"THENRETURN
3520 :IF NMBR<>OTPT THEN TC=TC+8:GOTO3550
3530 PRINT"CORRECT":AS=1:GOSUB9000:NEXT Q:FF=1
3540 GOSUB6800GET TREASURE:GOTO3590
3550 FORI=1TO500:NEXT:PRINT"SORRY, YOU DIDN'T GET IT"
3560 IF NT=1 THEN PRINT"BUT THE TREASURE HAS BEEN TAKEN, ANYWAY!":GOTO3580
3570 IF EARN>0 THENPRINT"YOU LOST A ";EARN$(EARN):EARN=EARN-1:GOSUB6400
3580 AS=10:GOSUB9000:X=0:Y=24:GOSUB8800:GOSUB8600
3590 TMT=TMT+1:RETURN
3600 REM**EVALUATE
3610 POKE CR,12:PRINT"
3620 IFTMT=0THEN3730
3630 AV=INT(10*EA-TC/TMT):IFAV>100THENAV=100
```

```
3640 IFTMT>3THEN3660
3650 PRINT"YOU ONLY TRIED"TMT"MACHINE!";:AV=84:IFTMT>1THENPRINT"S!"
3660 PRINT:PRINT"YOU RATE AS A ";
3670 RATE$="MASTER WIZARD"
3680 IFAV<96THEN RA$="WIZARD
3690 IFAV<60THEN RA$="JUNIOR WIZARD
3700 IFAV<40THEN RA$="NOVICE WIZARD"
3710 IFTMT<4THEN RA$="NOVICE WIZARD"
3720 PRINT RA$
3730 IF EARN<1 THEN 3770
3740 PRINT"YOU COLLECTED ";:IF EARN=1 THEN PRINT"THIS TREASURE:":GOTO3760
3750 PRINT"THESE TREASURES:"
3760 PRINT:FOR I=1 TO EARN:PRINT EARN$(I),:NEXT I:GOTO3780
3770 PRINT"     YOU DON'T HAVE ANY TREASURES"
3780 RETURN
3800 REM**RETURN FROM GUESS
3810 PRINT"";
3820 Q7=TABLE-1:Q8=OW:OW=1
3830 FOR TABLE=1 TO Q7
3840 :B$=STR$(INPT(TABLE))
3850 :IF TABLE=19 THEN PRINT"   INPUT | OUTPUT[redacted]
3860 :GOSUB3200
3870 NEXT TABLE:TABLE=Q7:OW=Q8
3880 IF OW=0AND(TABLE<19)THENGOSUB7000
3890 RETURN
4000 REM**FUNCTION
4010 Z=RND(-SEED)
4020 GOSUB5C30 PICK COMB.(F1)
4030 F2=INT(RND(1)+.5):REM TRANSPOSE
4040 F3=ASC(FUN$(F1)) :REM CONDITION
4050 F4=VAL(MID$(FUN$(F1),2,1)) :REM EQ'N 1
4060 F5=VAL(MID$(FUN$(F1),3,1)) :REM EQ'N 2
4070 IF(F2=1)ANDLEN(FUN$(F1))<>4THEN X=F4:F4=F5:F5=X
4080 ON F3-64 GOSUB4200,4400,4600,4800
4090 EQUAT=F5:IF CND=0 THEN EQUAT=F4
4100 IF CND=0 THEN Z=RND(1):REM CHANGE RND# CONSTANTS FOR EQ'N 2
4110 ON EQUAT GOSUB5000,5100,5150,5200,5300,5400,5500
4120 RETURN
4200 REM--COND=ANY
4210 CND=1
4220 RETURN
4400 REM--COND=EVEN
4410 CND=0:IF INT(INPT/2)=INPT/2 THEN CND=1
4420 RETURN
4600 REM--COND=>K
4610 K0=INT(RND(1)*5)+5:CND=0:K=K0:IF INPT>K0 THEN CND=1
4620 RETURN
4800 REM--COND=FACTOR?
4810 N0=INT(RND(1)*3)+3
4820 CND=0
4830 IFINT(INPT/N0)=INPT/N0 THEN CND=1
4840 RETURN
5000 REM--Y=MX+B
5010 M=INT(RND(1)*4)+1
5020 B=INT(RND(1)*4)+1
5030 GOTO5180
5100 REM--LEVEL II
5110 M=INT(RND(1)*3)*2+1
5120 B=INT(RND(1)*3)*2+2
5130 GOTO5180
```

```
5150 REM--LEVEL III
5160 M=INT(RND(1)*5)-2:IFM=0THEN5160
5170 B=INT(RND(1)*5)-2:IFB=0THEN5170
5180 OTPT=M*INPT+B
5190 RETURN
5200 REM--Y=CONSTANT
5210 K0=INT(RND(1)*16)+5
5220 OTPT=K0
5230 RETURN
5300 REM--Y=K-X
5310 K0=INT(RND(1)*21)+10
5320 OTPT=K0-INPT
5330 RETURN
5400 REM--Y=AX2+B
5410 A=1
5420 B=INT(RND(1)*3)+1:GOTO5530
5500 REM--LEVEL II
5510 A=INT(RND(1)*2)+1
5520 B=INT(RND(1)*5)-2:IFB=0THEN5520
5530 OTPT=A*INPT*INPT+B
5540 RETURN
5800 REM--DIFFICULTY SORT
5810 ON MACH GOSUB5840,5860,5880,5900
5820 GOTO5910
5830 REM--LVL 1
5840 LOW=0:HIGH=3:RETURN
5850 REM LVL 2
5860 LOW=4:HIGH=8:RETURN
5870 REM LVL 3
5880 LOW=9:HIGH=15:RETURN
5890 REM LVL 4
5900 LOW=16:HIGH=20:RETURN
5910 REM RND PICK FROM CORRECT LVL
5920 F1=INT((HIGH-LOW+1)*RND(1))+LOW
5930 RETURN
6000 REM**OPENING GRAPHICS
6010 GOSUB12000:Y=21:GOSUB8800:FORI=1TO39:PRINT" ";:NEXT:A$=9:GOSUB9000
6020 X=14:Y=20:GOSUB8800:FORI=1TO16:PRINT"
6030 PRINT"
6040 B$="":X=15:Y=20:GOSUB8800:IFSNDTHENPOKED1,D4:POKED2,D5
6050 FORI=48TO63:A$=CHR$(I+15*(I=58)+14*(I=59)):FORJ=1TO8
6060 PRINTA$A$A$A$A$A$B$A$A$A$A$A$A$B$;:POKED3,(S-(I-48)*10)-10*J
6070 IF FNES(C)THENPOKED1,0:RETURN
6080 NEXT:IFI<63THENPRINTLEFT$(D$,16);
6090 NEXTI:POKED1,0:PRINT"
6100 A$=4:GOSUB9000:X=0:Y=23:GOSUB8800:GOSUB8600
6110 RETURN
6200 REM**PAINT MACHINE
6210 PRINT"YOU ARE NOW IN A":PRINT "PLACE$(1):PRINT"PLACE$(2)
6220 X=5:Y=6:GOSUB8800
6230 PRINT"
6240 PRINT"
6250 PRINT"
6260 PRINT"
6270 PRINT"
6280 PRINT"
6290 PRINT"
6300 PRINT"
6310 X=0:Y=7:GOSUB8800:PRINT"#";MID$(STR$(SE),2)
6320 RETURN.
```

```
6400 REM**TREASURE LIST
6410 IF EA=0THENRETURN
6420 PRINT"YOU NOW HAVE:"
6430 FORI=1TOEARN:IFI=EA ANDI>1THENPRINT"AND ";
6440 PRINT"A ";EARN$(I),:NEXT
6450 RETURN
6800 REM**GET TREASURE
6810 FORI=1TO1000:NEXT:PRINT"YOU GOT IT!!"
6820 IF NT=1 THEN 6870
6830 PRINTR$:R$;"THE TREASURE YOU HAVE EARNED FROM"R$"THE "PLACE$(1)" ";
6840 PRINTPLACE$(2)" IS"R$"A "TREASURE$"!"
6850 EARN=EARN+1:EARN$(EARN)=TREASURE$:IFEA>1THENGOSUB6400
6860 SEED(JRNY)=SEED:AS=9:GOSUB9000:GOTO6890
6870 PRINT"SOMEONE HAS ALREADY SOLVED THIS FUNCTION"
6880 PRINT"AND TAKEN THE TREASURE.":AS=10:GOSUB9000:TMT=TMT-1
6890 X=0:Y=23:GOSUB8800:GOSUB8600
6900 RETURN
7000 REM**FLASH BOX
7010 X=6:Y=9:GOSUB8800:AS=6:GOSUB9000
7020 PRINT"                 ████████████     █"B$;
7030 FORI=1TO6-LEN(B$):PRINT" █";:NEXT:PRINT"████████████               ";
7040 PRINT"";:GOSUB6200
7050 RETURN
7400 REM**NO TREASURE
7410 FORI=1TO20:IFSEED(I)=SEED THEN NT=1:I=20:NEXTI:RETURN
7420 NEXT I:NT=0
7430 RETURN
7600 REM**GET NUMBER
7610 NMBR$="":POKE158,0
7620 GOSUB8000:A=ASC(A$):IFA=13ANDNM$=""THEN7620
7630 IF LEN(NMBR$)>L THEN 7660
7640 IFA>44ANDA<58ANDA<>47THEN PRINTA$;:NMBR$=NMBR$+A$:GOTO7620
7650 IFA=81 OR A=209 OR A=192 THENPRINTLEFT$("███",LEN(NM$))"Q  ██";:NM$="Q"
7660 IF A=20 AND LEN(NM$)THENPRINTA$;:NMBR$=LEFT$(NM$,LEN(NMBR$)-1):GOTO7620
7670 IF A<>13 THEN 7620
7680 PRINT:NMBR=VAL(NMBR$)
7690 RETURN
7800 REM**YES/NO
7810 PRINT"(Y█ OR N█)? ";:YES=0
7820 GOSUB8000:IFA$=CHR$(13)ANDYETHEN7860
7830 IF(A$=" |"ORA$="Y")ANDYE=0THENYE=1:PRINTA$;
7840 IF(A$="/"ORA$="N")ANDYE=0THENYE=2:PRINTA$;
7850 GOTO7820
7860 RETURN
8000 REM**GET SINGLE CHAR
8010 GETA$:IFA$<>""THENPRINT" █";:GOTO8040
8020 CT=CT+1:IFCT>8THENPRINTMID$(" ▒",CX,1)"█";:CX=3-CX:CT=0
8030 GOTO8010
8040 IFA$=CHR$(3)THEN1190 END
8050 IFA$=CHR$(147)THEN SND=0
8060 IFA$=CHR$(19)THEN SND=1
8070 RETURN
8200 REM**GET STRING
8210 GOSUB8000:A=ASC(A$)
8220 IFB$=""ANDA=20THEN8210
8230 IFB$=""ANDA=13THENRETURN
8240 IFB$=""ANDA$="▒"THENB$="▒":RETURN
8250 IFB$=""ANDA$="-"THENB$="-":RETURN
8260 IFA=13THENAS=5:GOSUB9000:PRINTA$;:RETURN
8270 IFA=20ANDLEN(B$)<2THENPRINTA$;:B$="":GOTO8210
```

```
8280 IFA=20THENPRINTA$;:B$=LEFT$(B$,LEN(B$)-1):GOTO8210
8290 IFA<32OR(A>127ANDA<161)THEN8210
8300 IFLEN(B$)>18THEN8210
8310 PRINTA$;:B$=B$+A$
8320 IFA$=CHR$(34)THENPRINTA$"▮":
8330 GOTO8210
8600 REM**WAIT FOR RETURN
8610 IFPEEK(CR)-12THEN8630
8620 PRINT"TO GO ON, PRESS ▮RETURN▮ ";:GOTO8640
8630 PRINT"TO GO ON, PRESS ▮ ▮ ";
8640 POKE158,0:GOSUB8000:A=ASC(A$):ESC=0
8650 IFA=13THEN8680
8660 IFA$="-"THENESC=1:GOTO8680
8670 GOTO8640
8680 AS=5:GOSUB9000:RETURN
8800 REM**TO X,Y
8810 PRINT"▮":PRINT"▮";
8820 IFX=0ANDY=0THEN RETURN
8830 IFX=0THENPRINTLEFT$(D$,Y);:RETURN
8840 IFY=0THENPRINTTAB(X);:RETURN
8850 PRINTLEFT$(D$,Y)TAB(X);
8860 RETURN
9000 REM**AUDIO STUFF
9010 D1=59467:D2=59466:D3=59464:D4=16:D5=15:IF SND=0 THEN D4=0
9020 ON AS GOSUB9100,9200,9300,9600,9700,9800,9900,10000,10100,10200
9030 POKED1,0:POKED3,0:RETURN
9100 REM 1*RANDOM BEEPS
9110 FOR SI=1TO 3:D8=INT(RND(-RND(0))*200)+55
9120 POKED1,D4:POKED2,D5:POKED3,D8:FOR SJ=1TO60:NEXT
9130 NEXT:RETURN
9200 REM 2*WOBBLE
9210 POKED1,D4:POKED2,D5:FORSI=1TO10:FOR D8=250TO200STEP-5
9220 POKED3,D8:NEXT:FORD8=200TO250STEP5:POKED3,D8:NEXT
9230 NEXT:RETURN
9300 REM 3*WIZARD'S TUNE
9310 POKED1,D4:POKED2,D5:FORSJ=0TO1
9320 PRINT:PRINT"IF ";:S1=4:S2=188:GOSUB10300:IFFNES(C)THENRETURN
9330 PRINT"YOU ";:S1=2:S2=140:GOSUB10300:IFFNES(C)THENRETURN
9340 PRINTMID$("GO CAN.",1+3*SJ,3+SJ);
9350 S1=4:S2=115:GOSUB10300:IFFNES(C)THENRETURN
9360 PRINTMID$("INMAS",1+2*SJ,2+SJ);
9370 S1=4:S2=124:GOSUB10300:IFFNES(C)THENRETURN
9380 PRINTMID$("TO TER ",1+3*SJ,3+SJ);
9390 S1=4:S2=115:GOSUB10300:IFFNES(C)THENRETURN
9400 PRINT"THE ";:S1=4:S2=188:GOSUB10300:IFFNES(C)THENRETURN
9410 PRINT"WIZ";:S1=2:S2=140:GOSUB10300:IFFNES(C)THENRETURN
9420 PRINT"ARD'S ";:S1=4:S2=115:GOSUB10300:IFFNES(C)THENRETURN
9430 PRINTMID$("TP",1+SJ,1);"OWER ":PRINT
9440 S1=2:S2=124:GOSUB10300:IFFNES(C)THENRETURN
9450 PRINT"YOU";:PRINTMID$("'D 'LL ",1+3*SJ,3+SJ);
9460 S1=4:S2=188:GOSUB10300:IFFNES(C)THENRETURN
9470 PRINTMID$("BETCAR",1+3*SJ,3);
9480 S1=4:S2=140:GOSUB10300:IFFNES(C)THENRETURN
9490 PRINTMID$("TER RY ",1+4*SJ,4-SJ);
9500 S1=4:S2=124:GOSUB10300:IFFNES(C)THENRETURN
9510 PRINTMID$("NOT SOME ",1+4*SJ,4+SJ);
9520 S1=4:S2=115:GOSUB10300:IFFNES(C)THENRETURN
9530 PRINTMID$("GO TREA",1+3*SJ,3+SJ);
9540 S1=2:S2=124:GOSUB10300:IFFNES(C)THENRETURN
9550 PRINTMID$("ASURES ",1+SJ,1+5*SJ);
```

```
9560 S1=4:S2=115:GOSUB10300:IFFNES(C)THENRETURN
9570 PRINTMID$("LONE!HOME!",1+5*SJ,5);
9580 S1=2:S2=140:GOSUB10300:IFFNES(C)THENRETURN
9590 FORSI=1TO300:NEXT:PRINT:NEXT:RETURN
9600 REM 4*FLASH/BUZZ
9610 POKE 59464,255:POKE 59466,15
9620 FOR VI=1TO60:POKE59468,14+180*SND:POKE59468,12:NEXT
9630 POKE 59466,0:RETURN
9700 REM 5*RANDOM BEEP
9710 FORSJ=1TOINT(RND(-0)*5)+3:S2=INT(RND(1)*200)+55:S1=INT(RND(1)*30+30)
9720 POKE D1,D4:POKE D2,D5:POKE D3,S2:FOR SI=1TOS1:NEXT:POKED1,0:NEXT
9730 RETURN
9800 REM 6*ZAP
9810 POKED1,D4:POKED2,D5
9820 FORSI=10TO255STEP10:POKED3,SI:NEXT
9830 POKED1,0:POKED3,0:RETURN
9900 REM 7*ONE-SHOT SIREN
9910 POKED1,D4:POKED2,D5
9920 FORSI=150TO50STEP-2:POKED3,SI:NEXT
9930 FORSI=50TO150STEP2:POKED3,SI:NEXT
9940 RETURN
10000 REM 8*JOURNEY
10010 POKED1,D4:POKED2,D5:FORSI=255TO20STEP-2
10020 POKED3,SI:POKED3,SI-10:POKED3,SI-20:IF FNES(C)THENRETURN
10030 NEXT:RETURN
10100 REM 9*FANFARE
10110 POKED1,D4:POKED2,D5:IFFNES(C)THENRETURN
10120 S1=8:S2=237:GOSUB10310
10130 S1=8:S2=211:GOSUB10310
10140 S1=8:S2=188:GOSUB10310
10150 S1=4:S2=158:GOSUB10310
10160 S1=8:S2=211:GOSUB10310
10170 S1=1:S2=158:GOSUB10310
10180 RETURN
10200 REM 10*UH-OH
10210 POKE D1,D4:POKE D2,D5
10220 S1=4:S2=90:GOSUB10310
10230 S1=2:S2=124:GOSUB10310
10240 RETURN
10300 S2=S2-30*SJ
10310 POKED3,S2:FORSI=1TO2*PEEK(151)/S1:NEXT:POKED3,0:FORSI=1TO5:NEXT:RETURN
11000 REM**TEST GRAPHIC
11010 PRINTTAB(8);"▒     █     █     █     "
11020 PRINTTAB(8);"▒██ ███ ██████ ████████ ██"
11030 PRINTTAB(8);"▒██ ███   ███      ███ ██"
11040 PRINTTAB(8);"▒██ ███ ██████████ ███ ██"
11050 PRINTTAB(8);"▒██ ███     █      ███ ██"
11060 PRINTTAB(10);"▒▒▒INPUT    |  OUTPUT▒████████████████████┼────────┼────────"
11070 RETURN
12000 REM**TITLE PAGE
12010 PRINTTAB(15)"▒▒▒◆♠▁▔▒▒"R$;TAB(6)"♠ PART OF THE ◤╱─┌╱  ╱NIT▒
12020 PRINT" BY THE ╲ATH ╱ETWORK ─URRICULUM ▔ROJECT▒
12030 PRINTTAB(16)"♥─♥ ╱▒███████▒4.0 |♠♥◤─▒
12040 PRINTTAB(8)" ╱SE ▒♥ |◤─| @█ TO ESCAPE.▒"R$;TAB(14)"─|2 SOUND.▒
12050 PRINT"  ╱SE ▒┤▁▁█ TO TURN SOUND OFF, ▒|┌╲▔█ FOR ON.▒▒▒
12060 PRINTTAB(6);:GOSUB8600:PRINT"▒";POKE CR,12
12070 RETURN
READY.
```

```
EASY 2.1 DCAT

READY.

 10 SYS1039:RUN
READY.


READY.

  3 VERS=2.1
  4 GOSUB11600:GOTO11200
  5 REMCOPYRIGHT SAN FRANCISCO STATE UNIVERSITY, 1981, ALL RIGHTS RESERVED
 10 PRINT"■";:IFYTHENFORMQ=1TOY:PRINT"■";:NEXTMQ
 11 RETURN
 12 GOSUB14:I$=B$:GOTO3000
 14 Q=Y:Y=20:GOSUB10:FORQJ=1TO2:FORQZ=1TO39:PRINT" ";:NEXT:PRINT:NEXT
 15 PRINT"■■■":Y=Q:RETURN
200 REM-NETIN-
210 I$="":D1$="R"
220 GOSUB400:IF(PEEK(SQ)ANDRF)<>1THEN220
230 SYSRX:A$=CHR$(PEEK(RB)):IFA$=RT$THENPRINT"■ ■";:SYSSD:RETURN
240 I$=I$+A$:GOTO220
400 REM----BLINK CURSOR----
410 IFDTHENPRINT"■"D1$"■";:D=FA:RETURN
420 PRINT"■"D1$"■";:D=TR:RETURN
600 REM----NET OUT---
610 SYSBS:D1$="S":FORI=1TOLEN(O$):GOSUB400:POKEXB,ASC(MID$(O$,I,1)):SYSTX
615 NEXT
620 POKEXB,13:SYSTX:PRINT"■ ■";:GOTO200
1000 REM FIND OUTPUT
1010 CD%=0:FORLN=0TONL:EE=0:I$=RC$(LN):GOSUB7000:IFEETHENEE=EE+128:GOTO1080
1020 IF(OU<>TR)AND(OU<>FA)THENEE=25+128:GOTO1080
1030 IFOU=0THEN1070
1040 CD%=CD%+1:I$=RX$(LN):GOSUB7000:IFCD%=1THENQO=OU
1050 IFOU<>QOTHENEE=130:GOTO1080
1060 IFEETHEN1080
1070 NEXTLN:IFCD%=0THENEE=130
1080 IFEEAND127THENGOSUB9000
1090 OU=QO:RETURN
1200 MA=30:OF=65:TR=-1:FA=0:MS=20:AE=.00009:HN=80:NL=-1
1240 DIM SK$(MS),SK(MS),KEY$(MA),P2(10),P3(10),P4(10)
1250 DATA ↑,+,-,*,/,=,<,>,<=,>=,<>,AND,OR,LCM,GCD,RMD,RANDOM,FACT,MAXIMUM
1255 DATAMINIMUM,END,INPUT,ALWAYS,NEVER,END,SQR,NOT,ODD,EVEN,INT,END
1300 RS$="IJKLM":NK=0
1310 R2=NK+OF:GOSUB1400:R0=NK+OF:GOSUB1400:R1=NK+OF:GOSUB1400
1322 FORI=R2+13TOR0-1:R2$=R2$+CHR$(I):NEXTI
1324 FORI=R0TOR1-1:R0$=R0$+CHR$(I):NEXT:FORI=R1TOOF+NK-1:R1$=R1$+CHR$(I):NEXT
1330 BS=1048:SQ=BS+15:RB=BS+17:XB=BS+18:TX=BS+9:RX=BS+6:RF=1:SD=BS+3
1340 ML=5:DIMCO$(ML),EX$(ML),RC$(ML),RX$(ML):RT$=CHR$(13):ES$="—":AC$="▒"
1390 RETURN
1400 READKE$(NK):IFKE$(NK)<>"END"THENNK=NK+1:GOTO1400
1410 RETURN
1600 REM GETNUM
1605 B$=""
1610 IFLEN(B$)>4THENA=20:GOTO1630
1620 GOSUB1700:IF(A>47ANDA<58)ORA=45ORA=46THENPRINTA$;:B$=B$+A$:GOTO1610
1630 IFA=20ANDLEN(B$)THENPRINT"■";:B$=LEFT$(B$,LEN(B$)-1):GOTO1610
```

```
1640 IFA$=ES$ORA$=AC$THENB$=A$:RETURN
1650 IFA<>13ORLEN(B$)=0THEN1610
1660 IN=VAL(B$):RETURN
1700 REM GETCHAR
1705 T=TI
1710 IFTI<T+30 THEN PRINT"▒▮";:GOTO1740
1720 IFTI<T+60 THEN PRINT" ▮";:GOTO1740
1730 T=TI
1740 GETA$:IFA$=""THEN1710
1750 PRINT" ▮";:A=ASC(A$):RETURN
1800 REM GETNONEMPTY
1805 GOSUB1700:IFB$=""ANDA=20THEN1800
1810 IFA$=ES$ORA$=AC$THENB$=A$:RETURN
1820 IFLEN(B$)=LSANDA<>13ANDA<>20THEN1800
1830 IFA=13THENRETURN
1840 IFA=20ANDLEN(B$)<2THENPRINTA$;:B$="":GOTO1800
1850 IFA=20THENPRINTA$;:B$=LEFT$(B$,LEN(B$)-1):GOTO1800
1860 IFA<32OR(A>127ANDA<161)THEN1800
1870 PRINTA$;:B$=B$+A$:IFA=34THENPRINTCHR$(34)+"▮";
1880 GOTO1800
1900 IF(V1<1)OR(V2<1)THENRE=FA:RETURN
1910 IF(V2-INT(V2))>AETHENRE=FA:RETURN
1920 IF(V1-INT(V1))>AETHENRE=FA:RETURN
1930 EE=0:RETURN
1940 IFV1<0THENRE=FA:RETURN
1950 GOTO1920
1960 IF(V2<>-1)ANDV2THENRE=FA:RETURN
1970 IF(V1<>-1)ANDV1THENRE=FA:RETURN
1980 EE=0:RETURN
2000 SK$(SS)=SK$:SS=SS+1:GOTO400
2200 SS=SS-1:SK$=SK$(SS):GOTO400
2400 SK(SN)=SK:SN=SN+1:GOTO400
2600 SN=SN-1:SK=SK(SN):GOTO400
2800 P0$=P1$:IFP1>LEN(I$)THENP1$="":GOTO2990
2840 IFP1>LEN(I$)THENP1$="":GOTO2990
2850 IFMID$(I$,P1,1)=" "THENP1=P1+1:GOTO 2840
2860 P1$=MID$(I$,P1,1):P1=P1+1
2990 RETURN
3000 GOSUB3200:GOSUB3400:IFNOTREORERTHEN3190
3035 IF P1$<>""THENRE=FA:GOTO3190
3040 GOSUB2200:O$=SK$:PRINT"■ ▮";
3190 RETURN
3200 P0=1:P1=1:GOSUB2800:ER=FA:RE=TR:SS=0:SN=0:O$="":LV=0:OV=11:D1$="T":RETURN
3400 LV=LV+1:IFLV>OVTHEN3800
3410 GOSUB3600:IFNOTRETHEN3590
3430 GOSUB5200:IFNOTRETHENRE=TR:GOTO3590
3450 GOSUB3600:IFNOTRETHENRE=FA:ER=TR:GOTO3590
3470 GOSUB2200:E1$=SK$:GOSUB2200:R$=SK$:GOSUB 2200:E2$=SK$
3500 SK$=R$+" "+E1$+" "+E2$:GOSUB2000:GOTO3430
3590 GOTO3900
3600 LV=LV+1:IFLV>OVTHEN3800
3610 P2(LV-2)=1:GOTO3800
3620 IFNOTRETHEN3790
3630 GOSUB5000:IFNOTRETHENRE=TR:GOTO3790
3650 P2(LV-2)=2:GOTO3800
3660 IFNOTRETHENRE=FA:ER=TR:GOTO3790
3670 GOSUB2200:T$=SK$:GOSUB2200:R$=SK$:GOSUB2200:E$=SK$
3700 SK$=R$+" "+E$+" "+T$:GOSUB2000:GOTO3630
3790 GOTO3900
3800 REMTERM
```

```
3810 P3(LV-2)=1:GOTO4000
3820 IFNOTRETHEN3990
3830 GOSUB4800:IFNOTRETHENRE=TR:GOTO3990
3850 P3(LV-2)=2:GOTO4000
3860 IFNOTRETHENRE=FA:ER=TR:GOTO3990
3870 GOSUB2200:F$=SK$:GOSUB2200:R$=SK$:GOSUB 2200:T$=SK$
3900 SK$=R$+" "+T$+" "+F$:GOSUB2000:GOTO3830
3990 ONP2(LV-2)GOTO3620,3660
4000 REMFACTOR
4010 GOSUB5000:SK=RE:GOSUB2400
4015 P4(LV-2)=1:GOTO4200
4020 IFERTHEN4150
4040 SK=RE:GOSUB2400
4050 GOSUB5300:IFNOTRETHENRE=TR:GOTO4080
4055 P4(LV-2)=2:GOTO4200
4060 IFNOTRETHENER=TR:GOTO4150
4074 GOSUB2200:F2$=SK$:GOSUB2200:P$=SK$:GOSUB2200:F1$=SK$
4079 SK$=P$+" "+F1$+" "+F2$:GOSUB 2000:GOTO4050
4080 GOSUB2600:IFNOTSKTHEN4150
4100 GOSUB2600:IFNOTSKTHEN4150
4120 GOSUB2200:F$=SK$:GOSUB2200:R$=SK$:SK$=R$+" 0 "+F$:GOSUB2000
4150 ONP3(LV-2)GOTO3820,3860
4200 REMPRIMARY
4210 IFP1$="."OR(P1$>="0"ANDP1$<="9")THENGOSUB4600:GOTO4390
4220 IFP1$>="A"ANDP1$<="Z"THENGOSUB4400:GOTO4390
4230 IFP1$<>"("THENRE=FA:ER=TR:GOTO 4390
4240 GOSUB2800:GOSUB3400:IFNOTRETHENRE=FA:ER=TR:GOTO4390
4270 IFP1$<>")"THENRE=FA:ER=TR:GOTO 4390
4280 RE=TR:GOSUB2800
4390 ONP4(LV-2)GOTO4020,4060
4400 LV=LV+1:IFLV>OVTHEN8800
4410 NM$=""
4420 IFP1$<"A"ORP1$>"Z"THEN4440
4430 GOSUB2800:NM$=NM$+P0$:GOTO4420
4440 S$=R2$:GOSUB5600:IFRETHENGOSUB6200:GOTO4590
4450 S$=R1$:GOSUB5600:IFRETHENGOSUB6400:GOTO4590
4460 S$=R0$:GOSUB5600:IFNOTRETHENER=TR
4590 GOTO8900
4600 SK$=""
4620 IFP1$<"0"ORP1$>"9"THEN4700
4630 GOSUB2800:SK$=SK$+P0$:GOTO4620
4700 IFP1$<>"."THEN4780
4710 IFLEN(SK$)=0THENSK$="0"
4720 GOSUB2800:SK$=SK$+P0$
4730 IFP1$<"0"ORP1$>"9"THEN4780
4740 GOSUB2800:SK$=SK$+P0$:GOTO4730
4780 IFLEN(SK$)>0THENRE=TR:GOSUB2000:GOTO4790
4785 RE=FA
4790 RETURN
4800 IFP1$<>"*"ANDP1$<>"/"THENRE=FA:GOTO4990
4820 GOSUB2800:NM$=P0$:GOSUB6000:RE=TR:SK$=C$:GOSUB2000
4990 RETURN
5000 IFP1$<>"+"ANDP1$<>"-"THENRE=FA:RETURN
5020 GOTO4820
5200 NA=3:S$=RS$:GOSUB5400:IFRETHENRETURN
5230 IFP1$<>"<"ANDP1$<>"="ANDP1$<>">"THENRE=FA:RETURN
5240 GOTO4820
5300 IFP1$<>"↑"THENRE=FA:RETURN
5320 GOTO4820
5400 IFP1$=""THENRE=FA:GOTO 5590
```

```
5410 P0=P1:GOSUB2800:NM$=P0$:IFP1$=""THENRE=FA:P1=P0-1:GOSUB2800:GOTO5590
5420 GOSUB2800:NM$=NM$+P0$:CC=2:GOSUB5600:IFRETHEN5590
5430 GOSUB5600:IFRETHEN5590
5450 IFCC=NAORP1$=""THENRE=FA:P1=P0-1:GOSUB2800:GOTO5590
5460 GOSUB2800:NM$=NM$+P0$:CC=CC+1:GOTO5430
5590 RETURN
5600 GOSUB6000:IFNOTRETHEN5790
5640 FORC=1TOLEN(S$):RE=FA:IFMID$(S$,C,1)=C$THENRE=TR:C=LEN(S$)
5660 NEXTC
5670 IFRETHENSK$=C$:GOSUB2000
5790 RETURN
6000 RE=FA:FORC=0TONK-1:IFNM$=KE$(C)THENRE=TR:C$=CHR$(C+OF):C=NK-1
6010 NEXTC:RETURN
6200 IFP1$<>"("THEN6380
6220 GOSUB2800:GOSUB3600:IFNOTRETHEN6380
6230 IFP1$<>","THEN6380
6240 GOSUB2800:GOSUB3600:IFNOTRETHEN6380
6250 IFP1$<>")"THEN6380
6260 GOSUB2800:GOSUB2200:F$=SK$:GOSUB2200:F$=SK$+" "+F$
6280 GOSUB2200:SK$=SK$+" "+F$:GOSUB2000:GOTO6390
6380 ER=TR:RE=FA
6390 RETURN
6400 IFP1$<>"("THEN6580
6420 GOSUB2800:GOSUB3400:IFNOTRETHEN6580
6430 IFP1$<>")"THEN6580
6440 GOSUB2800:GOSUB2200:F$=SK$:GOSUB2200:SK$=SK$+" "+F$:GOSUB2000:GOTO6590
6580 ER=TR:RE=FA
6590 RETURN
7000 GOSUB7200:GOSUB7400:IFNOTRETHEN7190
7050 GOSUB 2600:OU=SK:PRINT"■ ■■";
7190 RETURN
7200 P1=1:D1$="T"
7220 IFP1>LEN(I$)THENPRINT"ERROR IN INITTING OF EVAL":STOP
7230 P1$=MID$(I$,P1,1):P1=P1+1:IFP1$=" "THEN7220
7235 ER=FA:RE=TR:SS=0:SN=0:RETURN
7400 RE=TR:C$=P1$:GOSUB8000:IFC$>="0"ANDC$<="9"THENSK=VAL(T$):GOTO7580
7440 IFASC(C$)>=R0ANDASC(C$)<R1THENGOSUB8400:GOTO7580
7460 SK$=C$:GOSUB2000:GOSUB7400:GOSUB2200:C$=SK$:IFNOTRETHEN7590
7500 GOSUB2600:V1=SK:IFASC(C$)>=R1THENGOSUB8600:GOTO7580
7530 SK$=C$:GOSUB2000:SK=V1:GOSUB2400:GOSUB7400:GOSUB2200:C$=SK$
7560 IFNOTRETHENGOSUB2600:GOTO7590
7570 GOSUB2600:V2=SK:GOSUB2600:V1=SK:GOSUB7600
7580 GOSUB2400
7590 RETURN
7600 ONASC(C$)-(R2-1)GOTO7607,7610,7620,7630,7640,7650,7660,7670,7680,7690
7606 ON ASC(C$)-(R2+9)GOTO7700,7710,7720,7730,7740,7750,7760,7770,7780,7790
7607 IFLOG(ABS(V1)+.1)*V2>=HNTHENEE=5:RE=FA:RETURN
7608 SK=V1↑V2:IF(V1=INT(V1))AND(V2=INT(V2))THENSK=INT(SK+.5)
7609 RETURN
7610 SK=V1+V2:RETURN
7620 SK=V1-V2:RETURN
7630 IFLOG(ABS(V1)+.1)+LOG(ABS(V2)+.1)>HNTHENEE=3:RE=FA:RETURN
7635 SK=V1*V2:RETURN
7640 IFV2=0THENEE=4:RE=FA:RETURN
7645 SK=V1/V2:RETURN
7650 SK=(V1=V2):RETURN
7660 SK=(V2<V1):RETURN
7670 SK=(V2>V1):RETURN
7680 SK=(V2<=V1):RETURN
7690 SK=(V2>=V1):RETURN
```

```
7700 SK=(V1<>V2):RETURN
7710 EE=23:GOSUB1960:IFEETHENRETURN
7715 SK=(V1ANDV2):RETURN
7720 EE=24:GOSUB1960:IFEETHENRETURN
7725 SK=(V1ORV2):RETURN
7730 EE=11:GOSUB1900:IFEETHENRETURN
7731 IFV1<V2THENV3=V2:GOTO7734
7732 V3=V1:V1=V2:V2=V3
7734 IFINT(V2/V1)<>V2/V1THENV2=V2+V3:GOTO7734
7736 SK=V2:RETURN
7740 EE=12:GOSUB1900:IFEETHENRETURN
7741 IFV1=V2ORV2=0THEN7746
7744 V3=INT(V1-INT(V1/V2)*V2+.5):V1=V2:V2=V3:GOTO7741
7746 SK=V1:RETURN
7750 IFV2=0THENRE=FA:EE=8:RETURN
7755 SK=V1-INT(V1/V2)*V2:RETUPN
7760 SK=INT(RND(1)*(V2-V1+1)+V1):RETURN
7770 EE=20:GOSUB1900:IFEETHENRETURN
7771 SK=(INT(V2/V1)=V2/V1):RETURN
7780 SK=V1:IFV2>V1THENSK=V2:RETURN
7782 RETURN
7790 SK=V1:IFV2<V1THENSK=V2:RETURN
7792 RETURN
7800 P0$=P1$
7820 IFP1>LEN(I$)THENP1$="":GOTO7990
7830 P1$=MID$(I$,P1,1):P1=P1+1
7990 RETURN
8000 T$=""
8030 IFP1$=" "THENGOSUB7800:GOTO8030
8040 IFP1$=" "ORP1$=""THEN8150
8050 GOSUB7800:T$=T$+P0$:GOTO8040
8150 IFP1$=" "THENGOSUB7800:GOTO8150
8190 RETURN
8400 ONASC(C$)-R0+1GOTO8420,8430,8440
8420 SK=IN:RETURN
8430 SK=TR:RETURN
8440 SK=FA:RETURN
8600 ONASC(C$)-R1+1GOTO8620,8630,8640,8650,8660
8620 IFV1<0THENRE=0:EE=6:RETURN
8625 SK=SQR(V1):RETURN
8630 EE=26:GOSUB1970:IFEETHENRETURN
8635 SK=NOT(V1):RETURN
8640 EE=18:GOSUB1940:IFEETHENRETURN
8641 IFINT(V1/2)=V1/2THENSK=FA:RETURN
8645 SK=TR:RETURN
8650 GOSUB8640:SK=NOTSK:IFEETHENEE=19
8655 RETURN
8660 SK=INT(V1):RETURN
8800 PRINT"THAT'S TOO COMPLICATED, LEVEL ="LV:ER=TR:RES=FA:LV=LV-1:RETURN
8900 LV=LV-1:RETURN
9000 EE=EEAND127:PRINT"I DON'T.KNOW WHAT TO DO"
9020 EE=EE-1:IFEE>5THENEE=EE-1
9025 IFEE>6THENEE=EE-2
9030 IFEE>8THENEE=EE-5
9035 IFEE>11THENEE=EE-2
9040 IFEE>10THEN9052
9050 ONEEGOSUB9100,9104,9108,9112,9118,9116,9200,9204,9208,9212
9051 GOTO9053
9052 ONEE-10GOSUB9216,9300,9304,9310,9315
9053 PRINT"[illegible]POOF!■"
```

```
9055 GOSUB9600
9060 RETURN
9100 IFCD%=0THENPRINT"NO";:GOTO9102
9101 PRINT"TWO";
9102 PRINT" CONDITIONS ARE TRUE":RETURN
9104 PRINT"*";
9105 PRINT" CAUSES TOO LARGE A NUMBER":RETURN
9108 PRINT"/";
9109 PRINT" CAUSES DIVISION BY ZERO":RETURN
9112 PRINT"↑";:GOTO9105
9116 PRINT"RND";:GOTO9109
9118 PRINT"SQR OF NEGATIVE NUMBER":RETURN
9200 PRINT"LCM";
9201 PRINT" OF NON-NATURAL NUMBER(S)":RETURN
9204 PRINT"GCD";:GOTO9201
9208 PRINT"ODD";
9209 PRINT" OF NON-WHOLE NUMBER":RETURN
9212 PRINT"EVEN";:GOTO9209
9216 PRINT"FACT";:GOTO9201
9300 PRINT"AND";
9301 PRINT" CONNECTS NON-TRUTH EXPRESSION(S)":RETURN
9302 PRINT" NON-TRUTH EXPRESSION(S)":RETURN
9304 PRINT"OR";:GOTO9301
9310 PRINT"A CONDITION MUST BE TRUE OR FALSE":RETURN
9315 PRINT"I CAN'T 'NOT' A NON-TRUTH EXPRESSION":RETURN
9600 Y=24:GOSUB10
9605 PRINT"TO GO ON PRESS RETURN ";:GOSUB1700:IFA<>13THEN9600
9610 PRINT"":FORQZ=1TO21:PRINT" ";:NEXTQZ:RETURN
9800 PRINT"  _     _":PRINT"   \\   //":PRINT"┌──┐  ┌──┐":PRINT"│┌──    ──┐│"
9805 PRINT"│ │      │ │"
9810 PRINT"│└──    ──┘│":PRINT"└──┐  ┌──┘":PRINTTAB(4);"\  /":RETURN
9900 PRINT"";:GOSUB9990:PRINT"";:GOSUB9990:PRINT"";:GOTO9990
9990 PRINT"      ":RETURN
9999 GOSUB9900:Y=20:GOSUB10:PRINTTAB(12);:GOSUB1000:GOSUB9800:RETURN
10000 GOSUB11800
10008 FORQQ=0TO4
10010 B$=CO$(QQ):O$=RC$(QQ):Y=QQ*4:QL$=STR$(QQ+1)+". CONDITION: "
10015 QF=1:GOTO10050
10020 B$=EX$(QQ):O$=RX$(QQ):Y=QQ*4+2:QL$="    OUTPUT = ":QF=2:GOTO10050
10030 NEXTQQ:GOSUB11950:GOTO10000
10050 GOSUB10:PRINTQL$:B$;:X$=B$:LS=63:GOSUB1800
10055 IFB$=""ORRIGHT$(B$,1)="?"THEN10050
10060 IFB$=ES$THENGOSUB11900:GOTO10000
10065 IFB$=AC$THENGOSUB11950:GOTO11200
10070 IFB$=X$THEN10080
10075 GOSUB12
10077 IFERORRE=0THENB$=B$+"?":PRINT"I DON'T UNDERSTAND":GOTO10050
10080 IFQF=2THEN10090
10085 CO$(QQ)=B$:RC$(QQ)=O$:GOTO10020
10090 EX$(QQ)=B$:RX$(QQ)=O$:GOTO10030
10100 IFNL<0THENPRINT"I DON'T HAVE A MACHINE TO RUN":GOSUB9600:GOTO11200
10101 QG=0:PRINT"";:GOSUB9800
10103 Y=22:GOSUB10:PRINT"TYPE SHIFT  TO SEE THE MENU"
10104 PRINT"TYPE SHIFT  TO TEST YOURSELF"
10105 GOSUB14:Y=19:GOSUB10:PRINT:PRINT"INPUT: ";:GOSUB1600
10106 IFB$=ES$THEN11200
10107 IFB$=AC$THEN11100
10110 GOSUB9999:IFEE=0THENGOSUB11000:QG=QG+1:IFQG=19THENQG=0
10120 GOTO10105
11000 PRINT"";
```

```
11010 IFQQ=0THENPRINT"INPUT OUTPUT                    ":QQ=1:GOTO11000
11020 FORQZ=1TOQQ+1:PRINT" ";:NEXTQZ:PRINTIN;:QZ=LEN(STR$(IN))
11030 FORQJ=1TOQZ:PRINT" ";:NEXTQJ:PRINT"          ";OU:RETURN
11100 REM TEST
11103 PRINT" ";TAB(18);"TEST"
11105 FORQQ=1TO5:Y=RND(RND(0)):QR=INT(RND(1)*50+1)
11110 PRINT"IF INPUT = ";QR;
11120 PRINTTAB(15)"THEN OUTPUT = ";:GOSUB1600:PRINT" ":IFB$=ES$THENGOTO11200
11125 I9=IN:IN=QR:GOSUB1000
11130 IFEETHENQQ=QQ-1:GOTO11200
11140 IFABS(I9-OU)>AETHENPRINT:PRINT"THAT'S NOT WHAT I GET":GOSUB9600:GOTO10100
11150 NEXTQQ:PRINT:PRINT"I THINK YOU'VE GOT IT!!"
11160 GOSUB9600:GOTO11200
11200 REMMENU
11205 PRINT" ";TAB(12);"EASY SPEAK MENU
11207 PRINT"YOU CAN:":QM$=" MACHINE"
11210 PRINT"A) CREATE A";QM$:PRINT"B) LOAD A";QM$" FROM THE NETWORK
11215 PRINT"C) SEE NAMES OF"QM$"S ON NETWORK
11217 PRINT"D) DELETE"QM$" FROM NETWORK
11218 PRINT"E) QUIT
11220 IFNL<0THENGOTO11300
11230 PRINT"F) RUN THE";QM$:PRINT"G) TEST YOURSELF ON THE";QM$
11240 PRINT"H) EDIT THE";QM$:PRINT"I) SAVE THE";QM$;" ON NETWORK"
11300 PRINT:PRINT"YOUR CHOICE: ";:GOSUB1700:IFA<65ORA>73THEN11200
11310 IFNL<(A-64)THEN11200
11320 QJ=A-64:PRINT" ";A$;:GOSUB1700:IFA<>13THEN11200
11330 ONQJGOTO11500,12000,12600,12900,13000,10100,11100,10000,12400
11400 REMCLRPROG
11410 FORQZ=0TO4:CO$(QZ)="":EX$(QZ)="":NEXTQZ:NL=-1:RETURN
11500 REMCREATE
11510 GOSUB11400:GOTO10000
11600 REM-TITLE
11602 PRINT" ";:POKE59468,14
11610 PRINTTAB(14)"
11615 PRINTTAB(5)"
11620 PRINT"
11630 PRINTTAB(19)"
11640 PRINTTAB(16)"4.0
11660 GOSUB1200:GOSUB9600:PRINT" ":POKE59468,12
11670 RETURN
11700 PRINT" ":FORQQ=0TONL:Y=QQ*4:GOSUB10:PRINTQQ+1;"  CONDITION: ";CO$(QQ)
11710 Y=Y+2:GOSUB10:PRINT"     OUTPUT = ";EX$(QQ):NEXTQQ:RETURN
11800 GOSUB11700:Y=22:GOSUB10:PRINT"TYPE SHIFT 2 TO ERASE A STATEMENT"
11810 PRINT"TYPE SHIFT 7 WHEN FINISHED":RETURN
11900 FORQZ=QQTO3:CO$(QZ)=CO$(QZ+1):EX$(QZ)=EX$(QZ+1)
11902 RC$(QZ)=RC$(QZ+1):RX$(QZ)=RX$(QZ+1):NEXTQZ
11905 CO$(4)="":EX$(4)="":RC$(4)="":RX$(4)=""
11910 GOSUB11950:RETURN
11950 FORQZ=0TO4
11955 NL=QZ-1:IFCO$(QZ)=""OREX$(QZ)=""THENCO$(QZ)="":EX$(QZ)="":RETURN
11960 NEXTQZ:NL=4:RETURN
12000 REM-NETLOAD
12002 PRINT" NETLOAD":GOSUB12800:IFA$=ES$THEN12130
12010 PRINT:PRINT"NAME OF MACHINE":PRINT"TO LOAD: ";
12030 B$="":LS=16:GOSUB1800:NM$=B$:IFB$=ES$THEN12130
12040 O$="DER"+NM$:GOSUB600:Y$=I$
12060 IF LEFT$(I$,4)<>"DERT"THENPRINT:PRINT"NOT FOUND":GOTO12010
12070 GOSUB11400
12080 GOSUB12200:IF ER THEN 12010
12085 IF RES THEN 12130
```

```
12090 NL=NL+1:CO$(NL)=B$:GOSUB12:RC$(NL)=O$:GOSUB12200:IFERTHEN12010
12110 EX$(NL)=B$:IFRETHENPRINT:PRINT"NET ERROR":GOTO12010
12115 GOSUB12:RX$(NL)=O$:GOTO12080
12130 GOTO11200
12200 REM-GETLINE
12210 RES=FA:ER=FA:X$=MID$(Y$,5,4):IFX$="0000"THENRE=TR:RETURN
12230 O$="DES"+X$:GOSUB600:Y$=I$:IFLEFT$(I$,4)<>"DEST"THENPRINT:PRINT"NET ERROR
12235 B$=MID$(Y$,9):RETURN
12400 REM-NETSAVE
12402 PRINT"■■NETSAVE":GOSUB12800:IFA$=ES$THEN12540
12410 PRINT:PRINT"■NAME OF MACHINE: ";
12420 B$="":LS=16:GOSUB1800:SN$=B$:IFB$=ES$THEN12540
12430 PRINT:PRINT"YOUR NAME: ";
12440 B$="":LS=16:GOSUB1800:PRINT:NM$=B$:IFB$=ES$THEN12540
12450 GOSUB13200:O$="DEA"+O$+NM$+SN$:GOSUB600
12470 IF LEFT$(I$,4)<>"DEAT"THENPRINTSN$" IN USE!":GOTO12410
12475 B$=MID$(I$,5):FORN=0TONL
12490 O$="DEL"+B$+CO$(N):GOSUB600
12500 IF LEFT$(I$,4)<>"DELT"THEN PRINT "NET ERROR":N=NL:GOTO12530
12510 O$="DEL"+B$+EX$(N):GOSUB600
12520 IFLEFT$(I$,4)<>"DELT"THEN PRINT "NET ERROR":N=NL:GOTO12530
12530 NEXT
12540 GOTO11200
12600 REM NETNAMES
12602 PRINT"■■NETNAMES":GOSUB12800:IFA$=ES$THEN12670
12610 PRINT"■BEGINNING WITH: ";:B$="":LS=16:GOSUB1800:SK$=B$:PRINT
12615 IFA$=ES$THEN12670
12620 O$="DEN000"+SK$
12630 GOSUB600:IFMID$(I$,4,1)<>"T"THEN12660
12640 PRINTMID$(I$,8):GETA$:IFA$="—"THENGOTO12660
12650 O$="DEN"+MID$(I$,5,3)+SK$:GOTO12630
12660 GOSUB9600
12670 GOTO11200
12800 REM-DIAL
12810 B$="":PRINT:PRINT"ARE YOU CONNECTED TO THE NETWORK?
12815 PRINT"(■Y■, ■N■, OR ■SHIFT @■) ";:LS=1:GOSUB1800
12820 PRINT:IFLEFT$(B$,1)="Y"ORA$=ES$THENRETURN
12825 PRINT"DO YOU HAVE A MODEM? ";
12827 B$="":GOSUB1800:IFB$<>"Y"ORA$=ES$THENA$=ES$:RETURN
12828 PRINT
12830 PRINT"PLEASE DIAL 469-2126":O$="EASY SPEAK HELLO":GOSUB600:PRINTI$:RETURN
12900 REM-NETDEL
12902 PRINT"■■NETDELETE":GOSUB12800:IFA$=ES$THEN12990
12920 PRINT:PRINT"NAME OF MACHINE
12925 PRINT"TO DELETE: ";:B$="":LS=16:GOSUB1800:IFA$=ES$THENGOTO12990
12927 SN$=B$
12930 PRINT:PRINT"YOUR NAME: ";:B$="":LS=16:GOSUB1800:IFA$=ES$THEN12990
12935 NM$=B$:GOSUB13200:O$="DED"+O$+NM$+SN$:GOSUB600
12945 A$=MID$(I$,4,1)
12947 IFA$="C"THENPRINT:PRINT"■WRONG NAME OF CREATOR OF"QM$:GOTO12930
12950 IFA$="F"THENPRINT:PRINT"■NO MACHINE BY THAT NAME":GOTO12920
12955 IFI$="DEDT"THENPRINT:PRINT"■SUCCESSFUL DELETE
12960 GOSUB9605
12990 GOTO11200"DED"O$NM$SN$:600
13000 REMQUIT
13010 PRINT"■■■GOOD B-E THEN!":POKE40,1:POKE41,4:END
13200 REM FRMT O$
13210 O$=MID$(STR$(LEN(NM$)),2):IFLEN(O$)<2THENO$="0"+O$:RETURN
READY.
```