The Great Computer Challenge
Basic
Level IV

Problem 1: Popcorn Magic (20 points)

The local farm cooperative has hired you to produce a bar chart of gourmet-popcorn production on a farm by farm basis. The program should ask the user to enter the name of the farm, the number of acres, and the number of pints of popcorn produced. This information should be entered for each farm. After all the farms have been entered, the bar chart should be displayed on the screen. This output should consist of a single line for each farm, with the name of the farm, and a series of symbols representing the number of pint jars of popcorn per acre. The production goal for the year is 5000 jars per acre. A vertical bar should appear in the chart indicating this goal. A separate symbol should be used for the bars below and above this value. The following is provided as an example:

Input:

Orville's Acres, 144.8, 43801
Hoffman's Hills, 77.2, 36229
Jiffy Quick Farms, 89.4, 24812
Jolly Good Plantations, 183.4, 104570
Organically Grown Inc., 45.5, 14683

Output:

```
                    Popcorn Magic Inc.

    Production in Thousands of Pint Jars per Acre
-------------------------------------------------------
  Farm                    1   2   3   4   5   6   7
                         ---|---|---|---|---|---|---|
                                                |
Orville's Acres          ***************       |
Hoffman's Hills          ******************|
Jiffy Quick Farms        ***********           |
Jolly Good Plantations   ###################|###
Organically Grown Inc.   *************         |
-------------------------------------------------------
```

The Great Computer Challenge
Basic
Level IV

Problem 2: Star Gazing

You work for the Jet Propulsion Laboratory. They want you to write a program that takes an array containing the digitized representation of a picture of the night sky and located the stars on it. Each element of the array represents the amount of light hitting that portion of the image when the picture was taken. Intensities range from 0 through 20. The user is asked to enter the array of intensities. The following is provided as an example:

Input:

```
 0   3   4   0   0   0   6   8
 5  13   6   0   0   0   2   3
 0   8   0   0   0  17   3   2
 0   2   0   2  12  19  14   6
 0   0   0  12   0  14   2   3
 0   0  11  17  18   0   0   0
 0   0   0  13   0   0   0   0
```

Output:

The output should be a star map. A star is considered to be present if an element and its four adjacent elements have an average intensity of more than 6. In the matrix provided above, stars are located in locations (2,2), (4,6) and (6,4). The following is provided as a sample of the corresponding star map.

```
   ---1---2---3---4---5---6---7---8
   |
   |       *
   |
   |                         *
   |
   |                 *
```

The Great Computer Challenge
Basic
Level IV

Problem 3: Here come da judge (30 points)

The judges have decided to write a system which will keep track of all the students who enter the Great Computer Challenge. However, we desperately need a data entry screen design for the system. If you help us with our system, we'll help you win the contest. That seems fair enough.

Your program should display an entry screen just as it would look after one of the judges had entered all the information about Susan K. Smith, a contestant from one of the local high schools. Your program doesn't have to process or accept any data, it just has to display the screen design.

Your work will be judged on program structure and program clarity as always. You will also be judged on :

- Appropriate choice of data fields on the entry screen. Decide what kind of information we might like in our database.
- Clarity of the labels and headings
- How easy it would seem to be for a judge to have entered the information correctly–remember the judges are **not** rocket scientists.
- Pleasing and business-like appearance of your entry screen.
- Good use of function keys or mouse control.
- Whether it's easy to tell what was originally on the empty screen and what were the items that were entered by the (pretend) user.

Nothing in the display is actually suppose to work. You are simply showing us what a good entry screen would look like after a user had finished.

The Great Computer Challenge
Basic
Level IV

Problem 4: Loboc 2000 (again!) (30 points)

Anthropologist have discovered a primitive tribe in the Amazon call the Loboc, heretofore unknown to the outside world. In last year's programming contest, we described their problems with a calendar that was about to expire in the year 2000, leaving them unable to plant their crops. Fortunately, students in the Hampton Roads programming contest were able to help them, and the Loboc now have a bumper crop of *amendoim* (similar to giant peanuts, but served with a sauce of sautéed fire ants) that forms the staple of their diet. The anthropologists report that *amendoim* with fire ants– while an acquired taste–seems a lot like chicken. Local custom dictates that *amendoim* be planted from north to south in the following pattern:

North	**1**	**2**	**3**
	4	**5**	**6**
South	**7**	**8**	**9**

Each plant is 6 feet from the next (north-south and east-west). Of course, some plants dies before the harvest. The harvest is a big festival called *colheita,* and the surviving plants must be pulled in a special order dictated by the Great Electric Eel when the world began. The harvester must start in position **1** (whether there is a plant there or not); harvest each of the surviving plants (in any order the harvester chooses); and, after harvesting the last surviving *amendoim*, return to any one of the three planting positions (surviving plant or not) in the north row. This completes the *colheita.*

Your job is to provide the quickest (shortest path) for the harvester given the list of surviving plants. For example,

Blessing of the Great Electric Eel!
Which plants survive? 1, 3, 5, 8
The shortest path is: 1, 5, 8, 3
The length of the path is: 27.90

Your program may accept the list of surviving plants in a slightly different way, but if so, you need to make this clear to the user.

May the Great Electric Eel improve your programs and fill your hut with *amendoim* and fire ants!

The Great Computer Challenge
Basic
Level III

Problem 1: Parking is not Free (20 points)

Write a program that calculates parking fees for a multilevel parking garage. Ask whether the driver is an a car or a truck. Charge the driver $2 for the first hour, $3 for the second hour, and $5 for parking more than two hours. If the driver is in a truck, add an extra $1 to the total fee.

The Great Computer Challenge
Basic
Level III

Problem 2: Letter Magic (20 points)

Write a program using DO-LOOP UNTIL, that prints the ASCII characters from the number 65 to number 90 (uppercase letters A through Z)._ Immediately following that loop, print the characters backward, using a DO WHILE-LOOP statement.

The Great Computer Challenge
Basic
Level III

Problem 3: Making Money with Gadgets (30 points)

Write a program that will help a business person determine how many gadgets need to be sold before he/she can make a profit. The program must get the following information from the business person:

How much each gadget made will cost to make (ex: $3.00)
How much it will cost to buy a machine to make the gadget (ex: $50.00)
How much each customer will pay for each gadget (ex: $5.00)

The program will display to the screen, the data inputted by the business person, the number of gadgets that must be sold to make a profit (in the above example the answer is 25) and the total profit received from selling that many gadgets (in the above example the answer is $0).

The Great Computer Challenge
Basic
Level III

Problem 4: Graphic Gadgets (30 points)

Create a graphics program that will show a business person how much profit will be made from selling different numbers of gadgets.

Get the following information from the business person:

How much each gadget made will cost to make
How much it will cost to buy a machine to make the gadget
How much each customer will pay for each gadget

Create a graphic display that shows the profit made from 0 to 30 gadgets. If you can't create a graphic display, at least create a table that shows the same information (you will, however, receive less credit.)