



ECOO 2011

Programming Contest

Boardwide Contest

Not to be written before March 28, 2010

Problem 1 – Word Frame

The following program asks you to take a single word and use it to form a simple square frame, where the word is printed four times, once each for one of the four sides of the square: The corners and the centre of the square must contain stars. On top of the square the word is printed from left to right, containing a single space between letters as shown in the sample. On the bottom of the square the word is printed from right to left, also with spaces between letters. On the right the word reads from top to bottom without blank spaces and on the left the word reads from bottom to top.

DATA11.txt (DATA12.txt for the second try) contains 5 words on 5 separate lines. Each word contains less than 20 characters. Write a program to create a square as described above for each word. The program should pause between squares and proceed to the next one under input control

Sample Input:

```
CANADA
MAPLE
TO
FIRE
SHORT
```

Sample Output:

```
* C A N A D A *
A * * * * * C
D * * * * * A
A * * * * * N
N * * * * * A
A * * * * * D
C * * * * * A
* A D A N A C *
```

```
* M A P L E *
E * * * * * M
L * * * * * A
P * * * * * P
A * * * * * L
M * * * * * E
* E L P A M *
```

```
* T O *
O * * T
T * * O
* O T *
```

```
* F I R E *
E * * * * * F
R * * * * * I
I * * * * * R
F * * * * * E
* E R I F *
```

```
* S H O R T *
T * * * * * S
R * * * * * H
O * * * * * O
H * * * * * R
S * * * * * T
* T R O H S *
```


Problem 3 – Flip Cipher

Flip cipher scrambles the order of the characters in a message without changing their values. This is done by taking each character in turn from left to right and switching it with another character in the message. The character it will be switched with depends on a secret code word. In the example below the code word is “Wonderland”, where each letter stands for the position it holds within the alphabet. “Wonderland” then stands for the numbers: “23 15 14 4 5 18 12 1 14 4”

The first character of the message will therefore be switched with the character at position 1+23, the second character with the character at position 2+15, the third character with the character at position 3+14 and so on. If there is no character at a given position, then the procedure cycles through the start of the message.

The message “*But I didn't think.*” only has 19 characters, and so the “B” switches not with character 24 but character $24-19=5$, the “I”.

Successive switches will transform the message as follows:

Iut B didn't think.	Inu n.ditht B'itkd
Int B didn't thiuk.	Inu n.tithd B'itkd
Inu B didn't thitk.	Inu n. ithdtB'itkd
InuiB d dn't thitk.	Inu n. ithdtk'itBd
Inuin d dB't thitk.	'nu n. ithdtkIitBd
Inui nd dB't thitk.	'nu n. ithdtkiItBd
Inui n. dB't thitkd	'nu n. ithdtkiItBd
Inui n.d B't thitkd	'nu n. ithdtkiItdB
Inu n.diB't thitkd	'nu n. ithdtBiItkd
Inu n.dit't Bhitkd	

DATA31.txt (DATA32.txt for the second try) contains 6 lines. The first line contains the code word using capital letters. The next 5 lines contain encoded messages. Write a program that will DECODE messages that have been so encoded. Each encoded message contains less than 256 characters.

Sample Input

```
WONDERLAND
a.ua ltveehabn et cesg.e'i h l j etscce xIeethnaLnd
ase Stturot l?etl ladoreyeebuh
'nu n. ithdtBiItkd
dI, sTou.tk' tyt nsututtoeao thihnhui'njayol kth f nl.d ' s
a ' nee s k tntoaa hre cn,mv.gtdt o'IhiIoye
```

Sample Output

```
I have an excellent idea. Let's change the subject.
See all the trouble you started?
But I didn't think.
That's just it. If you don't think, then you shouldn't talk.
I've had nothing yet, so I can't take more.
```

Problem 4 – Perimeter Trees

Imagine a forest full of trees. Some trees are considered interior trees, and some are trees on the perimeter. A tree is considered interior, if for every imaginary straight line drawn through the tree, there will always be some trees on either side of the line. For a tree to be on the perimeter, a line may be drawn in such a way, that all trees are either on one side of the line or on the line itself. Given the x-y coordinates of a grove of trees, write a program that will count the number of trees that are on the perimeter.

DATA41.txt (DATA42.txt for the second try) contains 5 sets of data. The first line of each set contains an integer, n , where $n \leq 50$. It is followed by n lines containing two positive integers each, separated by a space, representing the x-y values of the position of the trees. Your task is to read this data and print the total number of trees that are on the perimeter of the grove

Sample Input:

8	79 27	39 85	3 16	44 75
98 27	74 53	15 7	53 88	60 8
6 47	75 16	46 32	24 34	9 19
93 3	3 61	48 2	88 16	95 11
35 39	75 6	43 79	71 52	31 55
100 73	40 79	2	59 54	93 33
96 43	89 40	70 81	88 100	61 84
68 64	72 93	27 83	38 100	
2 13	83 89	24	70 49	
11	34 79	76 90	43 46	
39 87	5	4 89	32 35	

Sample Output

```
There are 5 perimeter trees in this grove of 8 trees.  
There are 6 perimeter trees in this grove of 11 trees.  
There are 4 perimeter trees in this grove of 5 trees.  
There are 2 perimeter trees in this grove of 2 trees.  
There are 6 perimeter trees in this grove of 24 trees.
```