

You have to treat these questions as test questions and should solve them under close-book condition. You should not seek help in any form.

- Write a complete C program to code a price tag by alphabets. The simulated labels displayed on the screen are as follows:

```

C:\Windows\system32\cmd.exe
Enter price: $12.34
Coded price is $SR.ET

Do you have another price to enter <Y/N>: y
Enter price: $56.78
Coded price is $UP.MO

Do you have another price to enter <Y/N>: Y
Enter price: 90.12
Coded price is CK.SR

Do you have another price to enter <Y/N>: n
Press any key to continue . . . _

```

The mapping of the digits to alphabets is as follows:

Digit	0	1	2	3	4	5	6	7	8	9
Alphabet	X	S	R	E	T	U	P	M	O	C

- Write a complete C program to read a sentence of up to 200 characters from the keyboard and print on the screen the frequency of the small-letter alphabets in the string. For example:

```

C:\Windows\system32\cmd.exe
Type a sentence and press enter key: All of you are in the same boat.

Alphabet    Frequency
=====
a           3
b           1
e           3
f           1
h           1
i           1
l           2
m           1
n           1
o           3
r           1
s           1
t           2
u           1
y           1
Press any key to continue . . . _

```

- Two English words are anagram of each other if the letters from one word can be rearranged to form the other word. For examples, the following pairs of words are anagrams of each other: (dealer, leader), (there, three), (section, notices), (markers, remarks), (praised, despair), (teaching, cheating).

Write a complete C program to read two English words from the keyboard, and display on the screen whether the two words are anagrams. Assume that each word will not contain more than 15 alphabets and each alphabet in the word is small letter. The test run is as follows:

```

C:\Windows\system32\cmd.exe
Enter first word: madam
Enter second word: adam
"madam" and "adam" are not anagrams.

Enter * if you wish to stop:

Enter first word: salesmen
Enter second word: nameless
"salesmen" and "nameless" are anagrams.

Enter * if you wish to stop:

Enter first word: leader
Enter second word: dealer
"leader" and "dealer" are anagrams.

Enter * if you wish to stop:

Enter first word: markers
Enter second word: remark
"markers" and "remark" are not anagrams.

Enter * if you wish to stop: relayed

Enter first word: Enter second word: layered
"layered" and "layered" are not anagrams.

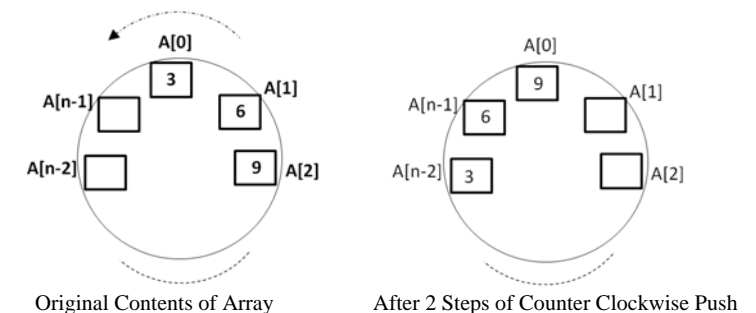
Enter * if you wish to stop:

Enter first word: three
Enter second word: there
"three" and "there" are anagrams.

Enter * if you wish to stop: *
Press any key to continue . . . _

```

- A 1-D array with n elements is linear, but it can be emulated by a cyclic structure as shown below:



Complete the following program to carry out the cyclic counter-clockwise push on the contents of the array until 0 is entered from the keyboard.

```
// rotate_counter.c      preparation for mock test
// Rotate counter-clockwise for a number of steps

#include <stdio.h>
#define n 10

void rotate_cc (int p[n], int count)
{
    // rotate counter clockwise for count steps
    :
    :
}

int main()
{
    int a[n], i, step;

    for (i=0;i<n;i++) a[i] = 3*i + 3;

    printf ("\n");
    for (i=0;i<n;i++) printf ("%d ",a[i]);
    :
    :
    return 0;
}
```

A session of the program execution is as follows:

```
C:\Windows\system32\cmd.exe
3 6 9 12 15 18 21 24 27 30
Enter the number of steps for counter clockwise push (0 to stop): 2
9 12 15 18 21 24 27 30 3 6
Enter the number of steps for counter clockwise push (0 to stop): 50
9 12 15 18 21 24 27 30 3 6
Enter the number of steps for counter clockwise push (0 to stop): 5
24 27 30 3 6 9 12 15 18 21
Enter the number of steps for counter clockwise push (0 to stop): 9
21 24 27 30 3 6 9 12 15 18
Enter the number of steps for counter clockwise push (0 to stop): 11
24 27 30 3 6 9 12 15 18 21
Enter the number of steps for counter clockwise push (0 to stop): 1
27 30 3 6 9 12 15 18 21 24
Enter the number of steps for counter clockwise push (0 to stop): 0
Press any key to continue . . .
```

- Write a complete C program to read up to 6 integers, and compute and print their least common multiple and highest common factor on the screen. The program runs continuously until there is no new set of integers to be processed. For examples:

```
C:\Windows\system32\cmd.exe
How many integers? 2
Enter the values: 20 6
Least Common Multiple: 60
Highest Common Factor: 2

Do you have another set of integers (Y/N): y
How many integers? 4
Enter the values: 20 30 40 90
Least Common Multiple: 360
Highest Common Factor: 10

Do you have another set of integers (Y/N): y
How many integers? 5
Enter the values: 62 62 62 62 62
Least Common Multiple: 62
Highest Common Factor: 62

Do you have another set of integers (Y/N): Y
How many integers? 3
Enter the values: 42 12 18
Least Common Multiple: 252
Highest Common Factor: 6

Do you have another set of integers (Y/N): Y
How many integers? 6
Enter the values: 5 7 11 13 17 19
Least Common Multiple: 1616615
Highest Common Factor: 1

Do you have another set of integers (Y/N): n
Press any key to continue . . .
```

Please solve the problems with
your best effort.

- A/Prof Tay