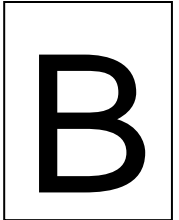


CONCORDIA UNIVERSITY
Winter 2012
Comp 248 /4 Section U - Object Oriented Programming I
Term Test #2/ B - SOLUTION



Question 1 - (10 points)

a) (5 pts.) What will the following code segment display?

```
int x = 0;
System.out.println( "Begin");
if (x <= 3)
if (x != 0)
System.out.println( "The second if");
else
System.out.println( "The else");
System.out.println( "Over");
System.out.println( "Begin again");
if (x > 3)
if (x != 0)
System.out.println( "The fourth if");
else
System.out.println( "Second else");
System.out.println( "Over again");
```

Output

```
Begin
The else
Over
Begin again
Over again
```

b) (2 pts.) What will the following Java code segments display on the screen? (Clearly mark each spaces by a ^).

```
boolean amount = false;
System.out.println("I don't want" +
((lots) ? "any " : "a lot ") + "of milk in my coffee");
```

Output

```
I^don't^wanta^lot^of^milk^in^my^coffee
```

c) (3 pts.) Write an if statement which will say "Valid" if the integer x is either a multiple of 6 or a multiple of 9 but not both and the message "Not valid" otherwise

One possible solution:

```
if(x%6 == 0 && x%9 != 0 || x%6 != 0 && x%9 == 0)
    System.out.println("Valid ");
else
    System.out.println("Not valid ");
```

One possible solution:

```
if((x%6 == 0 || x%9 == 0) && x%54 != 0)
    System.out.println("Valid");
else
    System.out.println("Not valid ");
```

There are many other ways of doing this.

Remember || results in true if one or both of conditions are satisfied.
We want only one condition.

Question 2 - (8 points)

a) (5 pts.) Given the program segment in the box, what is the output when

- i. ch1 = 'm' not uppercase so does not enter body of if
output is m
- ii. ch1 = 'Z' case 'Z' is executed
output is z (lower case)
- iii. ch1 = 'E' ch1--, is executed
output is D
- iv. ch1 = '2' not uppercase letter so does not enter body of if
output is 2
- v. ch1 = 'q' not uppercase letter so does not enter body of if
output is q

```
char ch1;
// ch1 is assigned a character
if (ch1 >= 'A' && ch1 <= 'Z')
{
    switch(ch1)
    {
        case 'A' :
        case 'E' :
        case 'U' : ch1--;
                    break;

        case 'Z' : ch1 = 'z';
                    break;

        default : ch1++;
                    break;
    }
}
System.out.print(ch1 );
```

b) (3 pts.) Rewrite the following code segment replacing the *switch* by a *nested if* statement.

```
if (ch1 >= 'A' && ch1 <= 'Z')
{
    switch(ch1)
    {
        case 'A' :
        case 'E' :
        case 'U' : ch1--;
                    break;
        case 'Z' : ch1 = 'z';
                    break;
        default : ch1++;
                    break;
    }
}
System.out.print(ch1 );
```

```
if (ch1 >= 'A' && ch1 <= 'Z')
{
    if(ch1 == 'A' || ch1 == 'E' || ch1 == 'U'
)
        ch1--;
    else if (ch1 == 'Z')
        ch1 = 'z';
    else
        ch1++;
}
System.out.print(ch1 );
```

Solution

Question 3. (6 points)

a) (3 pts.) Write the necessary Java statement(s) (not a complete program) to display on the screen in reverse order the content of a string variable `message`. Assume the variable is already declared and initialized. In other words if `message` contains the string `hello` there, your code segment should display `ereht olleh`.

```
for (int i = message.length-1; i >= 0; i--)
    System.out.print(message.charAt(i));
```

b) (3 pts) What will the following code segment display on the screen?

```
boolean more = true;
int firstInt = 5;
int secondInt = 21;
while (firstInt <= secondInt && more)
{
    if (secondInt / firstInt <= 2)
        more = false;
    else
        firstInt+=2;
}
System.out.println(more); // not in loop
System.out.println(firstInt);
```

Output

False
9

Question 4 - (6 pts)

What will the following code segment display on the screen?

```
int k=0, j=0, sum = 10;
for (k = 2; k < 8; k++)
for (j = 8; j > 2*k; j--=2)
{
System.out.print(" " + (k-j) + " +");
sum += (k-j);
}
System.out.println(" = " + sum);
System.out.println("k = " + k + " and j = " + j);
```

Trace: (using ^ for a space)

sum = 10
k = 2 (2 < 8)
 j = 8 (to j > 2*k = 4) output:^-6^+ sum = 10 + -6 = 4
 j = 6 (6 > 4 True) output:^-4^+ sum = 4 + -4 = 0
 j = 4 (4 > 4 False)
k = 3 (3 < 8)
 j = 8 (to j > 2*k = 6) output:^-5^+ sum = =0 + -5 = -5
 j = 6 (6 > 6 False)
k = 4 (4 < 8)
 j = 8 (to j > 2*k = 8, False)
k = 5 (5 < 8)
 j = 8 (to j > 2*k = 10, False)
k = 6 (6 < 8)
 j = 8 (to j > 2*k = 12, False)
k = 7 (7 < 8)
 j = 8 (to j > 2*k = 14, False)
k = 8 (8 < 8 - false)

<pre>-6 + -4 + -5 + = -5 k = 8 and j = 8</pre>	Output
--	---------------

Question 5 – (10 pts)

Write a complete Java program which will prompt the user for two dates and test whether the first date comes before the second date. Your program will output a message saying whether the first dates comes before the first date or not. The dates are entered as dd mm yyyy. Assume the dates entered are valid dates.

Here are a few sample runs to illustrate the expected behaviours (Bold Italic *numbers* are user input)

```
Enter 2 dates as dd mm yyyy dd mm yyyy: 14 10 2009 14 10 2009  
First date does not come before the second date
```

```
Enter 2 dates as dd mm yyyy dd mm yyyy: 14 10 2009 14 11 2009  
First dates comes before second date
```

```
Enter 2 dates as dd mm yyyy dd mm yyyy: 14 10 2009 10 09 2008  
First date does not come before the second date
```

One possible solution:

```
import java.util.Scanner;
public class Q5 {
    public static void main(String[] args) {
        int y1, m1, d1, y2, m2, d2;
        Scanner keyIn = new Scanner(System.in);

        System.out.print("Enter 2 dates as dd mm yyyy dd mm yyyy: ");
        d1 = keyIn.nextInt();
        m1 = keyIn.nextInt();
        y1 = keyIn.nextInt();
        d2 = keyIn.nextInt();
        m2 = keyIn.nextInt();
        y2 = keyIn.nextInt();

        if(y1 < y2 || (y1 == y2 && m1 < m2) || (y1 == y2 && m1 == m2 && d1 < d2))
            System.out.println("First date comes before the second date");
        else
            System.out.println("First date does not come before the second date");
    }
}
```