



*Philadelphia University*  
*Faculty of Engineering*

## **Marking Scheme**

Examination Paper

Department of CE

***Module: Programming Language (630203)***

First Exam

Second Semester

Date: 03/04/2011

Section 5

Weighting 15% of the module total

Lecturer:

Dr. Qadri Hamarsheh

Coordinator:

Dr. Qadri Hamarsheh

Internal Examiner:

Eng. Anis Nazer

## Marking Scheme Programming Language (630203)

The presented exam questions are organized to overcome course material, the exam contains 3 questions; *all questions* are compulsory requested to be answered. Thus, the student is permitted to answer any question out of the existing ones in this section.

### Marking Assignments

The following scheme shows the marks assignments for each question. They show also the steps for which a student can get marks along the related procedure he/she achieves.

**Question 1** This question is attributed with 6 marks if answered properly

The answer for this question as the following:

- 1) \_\_\_\_ is a valid **char** value.  
a. -129  
**b. 'A'**  
c. 128  
d. 129
- 2) Suppose that **alpha** and **beta** are **int** variables and  $\alpha = 5$  and  $\beta = 10$ . After the statement **alpha \*= beta;** executes, \_\_\_\_.  
a.  $\alpha = 5$   
b.  $\alpha = 10$   
**c.  $\alpha = 50$**   
d.  $\alpha = 50.0$
- 3) What is the value of **x** after the following statements execute?  

```
int x;  
x = (5 <= 3 && 'A' < 'F') ? 3 : 4
```

  
a. 2  
b. 3  
**c. 4**  
d. 5

- 4) Consider the following code.

```
int limit;  
int counter = 0;  
cin >> limit;  
while (counter < limit)  
{  
    cin >> entry;  
    triple = entry * 3;  
    cout << triple;  
    counter++;  
}  
cout << endl;
```

This code is an example of a(n) \_\_\_\_ **while** loop.

- a. flag-controlled  
**b. counter-controlled**  
c. EOF-controlled  
d. sentinel-controlled
- 5) Which of the following loops is guaranteed to execute at least once?  
a. counter-controlled while loop  
**c. do...while loop**  
b. for loop  
d. sentinel-controlled while loop
- 6) Suppose **j**, **sum**, and **num** are **int** variables, and the input is **26 34 61 4 -1**. What is the output of the code?

```
sum = 0;  
cin >> num;  
for (int j = 1; j <= 4; j++)  
{  
    sum = sum + num;  
    cin >> num;  
}  
cout << sum << endl;
```

- a. 124  
**b. 125**  
c. 126  
d. 127

**Question 2** This question is attributed with 4 marks if answered properly

The answer for this question as the following:

|    | C++ code   | output               |
|----|--|----------------------|
| 1) | <pre>int x = 35; int y = 45; int z; if (x &gt; y)     z = x + y; else     z = y - x; cout &lt;&lt; x &lt;&lt; " " &lt;&lt; y &lt;&lt; " " &lt;&lt; z &lt;&lt; endl;</pre>  | <pre>35 45 10</pre>  |
| 2) | <pre>char lastInitial = 'S'; switch (lastInitial) {case 'A':     cout &lt;&lt; "section 1" &lt;&lt;endl;    break; case 'B':     cout &lt;&lt; "section 2" &lt;&lt;endl;    break; case 'C':     cout &lt;&lt; "section 3" &lt;&lt;endl;    break; case 'D':     cout &lt;&lt; "section 4" &lt;&lt;endl;    break; default:     cout &lt;&lt; "section 5" &lt;&lt;endl;}</pre> | <pre>section 5</pre> |

**Question 3** This question is attributed with 5 marks, if answered properly.

The complete code for this question as the following:

```
#include <iostream>
using namespace std;
int main ()
{
    float fltNumber,avrPos,sumPos,sumNeg, avrNeg,counterPos, counterNeg;
    sumPos = 0.0;
    sumNeg = 0.0;
    counterPos =0;
    counterNeg = 0;
    cout << "Enter number, -999 to exit";
    cin >> fltNumber;
    while (fltNumber!=-999)
    {
        if (fltNumber < 0)
        {
            counterNeg++;
            sumNeg += fltNumber;
        }
        else
        {
            counterPos++;
            sumPos += fltNumber;
        }
        cout << "Enter number, -999 to exit";
        cin >> fltNumber;
    }
    if (counterNeg !=0)
    {
        avrNeg = sumNeg / counterNeg;
        cout << "the Average of entered negative numbers =" << avrNeg << endl;
    }
    else
        cout << "No Entry with negative numbers" << endl;
    if (counterPos !=0)
    {
        avrPos = sumPos / counterPos;
        cout << "the Average of entered positive numbers = " << avrPos << endl;
    }
    else
        cout << "No Entry with positive numbers" << endl;
    return 0;
}
```