Punjab University College of Information Technology (PUCIT), University of the Punjab

Programming Fundamentals – Spring 2012

Midterm Exam

Date: Monday, May 07, 2012

Time Allowed: **90 Minutes** Total Marks: **70**

Roll No. ______________________________

Student's Name: ______________________________

Student's Signature: ______________________________

Answer Book No:
Invigilator's Signature:

Tick Your Section

BSSE	***MOR***	
	AFT	
BSIT	***MOR***	
	AFT	
BSCS	***MOR-A***	***MOR-B***
	AFT-A	***AFT-B***

INSTRUCTIONS:

1. You are NOT allowed to ask any questions during the exam. If there is some mistake in the question paper, the benefit goes to the student.
2. Use of calculator is NOT allowed.
3. Use only blue or black ink to solve the exam. Even the lead pencil is also not allowed.
4. Suggested time is mentioned for each question. Manage your time accordingly.
5. Write your answers and working within the given space for each question.
6. You may use any blank/unused space on the exam booklet for rough work. No extra sheet will be provided for rough work.
7. Make sure that there are **8 double-side sheets** (16 pages) in this portion of the exam, including Title.
8. When you are told to open this booklet, first of all write your ROLL NUMBER on top of each sheet.
9. Do not forget to keep praying before and during the exam. And remain honest! ☺

Question #	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**TOTAL**
Total Marks	**15**	**15**	**10**	**10**	**10**	**10**	**70**
Marks Obtained							

DO NOT OPEN THIS BOOKLET

UNTIL YOU ARE TOLD TO DO SO!

☺ **Good Luck!** ☺

Examiner's Signature:

This page is left blank intentionally
(You may use it for your rough work)

QUESTION # 1 3x5 = 15 Marks Suggested Time: 20 minutes

Show the output for the following code segments. **Cutting/overwriting in any part of this question will result in ZERO marks in that part.** There is **NO syntax error** in any of these code segments.

a) **[3]**

```
int i = 5;
for(int i = 0; i < 2; i = i+1)
{
     cout << i << " ";
}
cout << i << " ";
cout << i++ << endl;
```

Output:

____

____

b) **[3]**

```
int number = 3;
int main()
{
     cout << number << " ";
     int number = 1;
     switch(number)
     {
          default:   cout << "0" << " ";
          case 1:    cout << "1" << " ";
          case 2:    cout << "2" << " ";
          case 3:    cout << "3" << " ";
     }
     return 0;
}
```

Output:

____

____

c) **[3]**

```
int fn (int n)
{
     return n-1;
}
int main()
{
     int num = 3;
     while(fn(num))
     {
          cout << num << " ";
          num = fn(num);
     }
     cout << num;
     return 0;
}
```

Output:

____

____

d)

```
void fn(int& num)                                              [3]
{
    num = num+1;
}
int fn(double num = 3)
{
    return num+2;
}
int main()
{
    int i;
    for(i = fn(); i<8; fn(i))
        cout << i << " ";
    return 0;
}
```

Output:

____

____

e)

```
int fn()                                                       [3]
{
     static int s = 0;
     s = s + 1;
     return s-1;
}
int main()
{
     int array1[4] = {1,2};
     int array2[] = {11,12};
     int index;

     for(int i = 0; i < 2; i=i+1)
     {
          index = fn();
          cout << array2[index] << " ";
     }
     for(; i < 4; i=i+1)
     {
          index = fn();
          cout << array1[index] << " ";
     }

}
```

Output:

____

____

QUESTION # 2 5+5+5 = 15 Marks Suggested Time: 15 minutes

Find **LOGICAL ERRORS**, if any, in the following parts (a) to (c)

(a) This program (Division.cpp) wants to display the result as per its **PURPOSE**. It **compiles correctly** but may not give desired output. You are required to give answer in the **ANSWER BOX**.

```
                              // Division.cpp
// PURPOSE: This program inputs numerator and denominator, and displays “Div-by-0 ERROR” if the value
// of denominator is ZERO otherwise it shows the message “Divisible”
#include <iostream>
using namespace std;
int main()
{
     int  num; // numerator
     int  den; // denominator
     // Now input the values
     cout << "Enter the numerator of division";
     cin >> num;
     cout << "Enter the denominator of division";
     cin >> den;
     // Now display appropriate message depending on the value of den
     cout<< "The result is: "
         << (den=0) ? "Divisible" : "Div-by-0 ERROR"
         << endl;
     return 0;
}
```

ANSWER BOX

[*Select only one option out of* ***A***, ***B***, ***C***]

A. The above code has **NO** logical error

B. The above code has logical error so **ALL** inputs will give UNDESIRED result

C. The above code has logical error so **SOME/MANY** inputs will give UNDESIRED result [following are **three** sample input sets]:

Sample 1		Sample 2		Sample 3	
num	**den**	**num**	**den**	**num**	**den**

(b) This program (CountByLoop.cpp) wants to display the result as per its **PURPOSE**. It **compiles correctly** but may not give desired output. You are required to give answer in the **ANSWER BOX**.

```
                                    // CountByLoop.cpp
// PURPOSE: This program inputs two values from user (start and end) and displays counting
// from start to end (both inclusive), using loop
#include <iostream>
using namespace std;
int main()
{
      int  start; // starting number for counting
      int  end;  // ending number for counting
      int  num;  // each individual number during counting
      // Now input the values
      cout << "Enter a number to start counting with ";
      cin >> start;
      cout << "Enter a number to end counting at ";
      cin >> end;
      // Now counting using function calls
      for (num=start; num <= end; ++num)
          cout << num++ << endl;
      return 0;
}
```

ANSWER BOX

[*Select only one option out of* ***A***, ***B***, ***C***]

A. The above code has **NO** logical error

B. The above code has logical error so **ALL** inputs will give UNDESIRED result

C. The above code has logical error so **SOME/MANY** inputs will give UNDESIRED result [following are **three** sample input sets]:

Sample 1		Sample 2		Sample 3	
start	**end**	**start**	**end**	**start**	**end**

(c) This program (F4Fun.cpp) wants to display the result as per its **PURPOSE**. It **compiles correctly** but may not give desired output. You are required to give answer in the **ANSWER BOX**.

```
                                        // F4Fun.cpp
// PURPOSE: This program inputs a value from user (seed) and displays next 4 numbers in counting
// from seed (1st value to be displayed is seed+1  and last value to displayed is seed+4), using function
#include <iostream>
using namespace std;

void fnShowInt(int seed)
{
      cout << seed++ << endl;
}
int main()
{
      int  seed; // seed for counting
      // Now input the value
      cout << "Enter the seed value";
      cin >> seed;
      // Now display counting starting from seed+1 till seed+4 using function calls
      fnShowInt(seed+1);
      fnShowInt(seed+2);
      fnShowInt(seed+3);
      fnShowInt(seed+4);
      return 0;
}
```

ANSWER BOX

[*Select only one option out of **A**, **B**, **C***]

A. The above code has **NO** logical error

B. The above code has logical error so **ALL** inputs will give UNDESIRED result

C. The above code has logical error so **SOME/MANY** inputs will give UNDESIRED result [following are **three** sample input sets]:

Sample 1	Sample 2	Sample 3
seed	**seed**	**seed**

This page is left blank intentionally
(You may use it for your rough work)

QUESTION # 3 10 Marks Suggested Time: 15 minutes

***TOPIC:* File processing, decision making, and simple looping**

You are given a file (**`students.txt`**) which contains records of several students. The first line of the input file contains an integer which indicates how many records of students are present in the file. After that, each line of the input file contains the roll number of a student, and his/her marks in the midterm exam of Programming Fundamentals ☺. There is a single space between the roll number and the marks of a student. See the sample input file shown below.

You are required to write a C++ program which reads the input file, and determines the **maximum marks obtained by a student of Morning shift** and the **maximum marks obtained by a student of Afternoon shift**. Your program should display this information on screen (see sample output shown below).

Sample input file:

students.txt

```
6
BITF11M340 59
BSEF11A932 61
BCSF11A356 70
BCSF11M201 65
BITF11A460 55
BSEF11M845 68
```

Sample output (on screen):

```
Max marks in Morning shift: 68
Max marks in Afternoon shift: 70
```

You can assume that all letters in the roll number are in capital case. You can also assume that all marks are positive integers.

Note: *Indent your code properly. Use meaningful variable names.*

You may continue the answer of this question on next page, if needed.

You may use empty space of this page for rough work, also.

QUESTION # 4 **10 Marks** **Suggested Time: 10 minutes**

***TOPIC:** Input validation, Nested loops*

Write a **C++ program** which prints the following pattern (cross) on screen. Size of the pattern should be taken from the user.

<u>Input validation:</u> *Your program should make sure that the **size** is an **odd number greater than or equal to 5**. In case of invalid input, your program should keep prompting the user again and again till the user provides valid input.*

Sample Run 1:

```
Enter the size: 3
Invalid input!! Enter again: 8
Invalid input!! Enter again: 7

*     *
 *   *
  * *
   *
  * *
 *   *
*     *
```

Sample Run 2:

```
Enter the size: 9

*       *
 *     *
  *   *
   * *
    *
   * *
  *   *
 *     *
*       *
```

IMPORTANT NOTE: Your program should use ONLY following THREE cout statements, in order to display the cross pattern on screen:

```
cout << "*";    cout << " ";    cout << endl;
```

You may continue the answer of this question on next page, if needed.

You may use empty space of this page for rough work, also.

QUESTION # 5 10 Marks Suggested Time: 15 minutes

***TOPIC:* Default arguments in functions / Function overloading**

Write **prototype** and **definition** of function(s) to correctly execute the given **`main`**.

> This function **`createLargest`** combines **two** or **three digits** in order to make the largest possible number. Each digit will be in the range from **0 to 9** (no need to validate). For example, if you have the three digits 2, 6, and 5, then the largest number that can be made from these digits is 652. If you have two digits 1 and 8, then the largest number will be 81.

Note: *Indent your code properly. Use meaningful variable names.*

```
#include <iostream>
using namespace std;
// Write prototype(s) of function(s) here



int main ()
{
    int num;
    createLargest (num, 1, 2);
    cout << num << endl;        // should display 21
    createLargest (num, 2, 6, 5);
    cout << num << endl;        // should display 652
    createLargest (num, 6, 0, 9);
    cout << num << endl;        // should display 960
    return 0;
}
// Write definition(s) of function(s) here
```

Continue the answer of this question on next page.

You may use empty space of this page for rough work, also.

QUESTION # 6 10 Marks Suggested Time: 10 minutes

***TOPIC:* Input c-string, and its processing**

Write a **C++ program** which takes a c-string from the user. After that, your program should take a **non-whitespace** (i.e. excluding space, tab, and newline) character from the user. Then, the program will count the occurrences of that character in the input string which was entered by the user. You are **NOT** allowed to include the header file **<cstring>** in your program, nor use any of its functions.

Sample Run 1:

```
Enter the string: A C++ program
Enter the character: a

The character a occurs 1 time(s) in the string
```

Sample Run 2:

```
Enter the string: The Midterm Exam
Enter the character: e

The character e occurs 2 time(s) in the string
```

You can assume that the maximum length of the input string entered by the user will be 40. The search should be case-sensitive e.g. ‘a’ and ‘A’ will be considered different characters.

You may continue the answer of this question on next page, if needed.

You may use empty space of this page for rough work, also.