UNIVERSITY REGULAR EXAMINATIONS

2013 /2014 ACADEMIC YEAR

1ST YEAR 2ND SEMESTER EXAMINATIONS

(MAIN EXAMINATION)

FOR THE DEGREE OF

BACHELOR OF SCIENCE (Computer Science, IT)

COURSE CODE: CSC 121

COURSE TITLE: PROCEDURAL PROGRAMMING

DATE: 22ND APRIL, 2014   TIME: 9:00A.M.- 12 NOON

Question 1 Compulsory (5 Marks each part for a total of 30 Marks)

a) Write a function that returns the smallest of its three numeric parameters.

b) What is the value in variable x after the following code has executed?

```
int x = 10;
for(;x < 40; x++) x = x + 10;
```

c) Write a function which exchanges the values inside its two variable actual parameters.

d) What is the value in variable x after the following code has executed?

```
float x;
x += 3.14;
```

e) What is the value in variable x and in variable y after the following code has executed?

```
float A[5] = {3.8, 5.6, 2.1, 3.3, 4.8, 6.3};
int x = A[1];
float y = A[5];
```

f) Assume the existence of the following function definitions.

```
int times2(int x) {return 2 * x; }
int sumof(int x, int y) {return x + y; }
int halfof(int x) {return x / 2; }
```

What is the value in variable x after the following code has executed?

```
int m = 5;
int n = 8;
int x = halfof(times2(sumof(sumof(2,times2(n)),halfof(m)))+3*n);
```

Question 2

a) Generally in C/C++, when we pass an array as a parameter to a function, we must also pass its size in another parameter.

i: Explain why. [5 Marks]

ii: Under what circumstance can we avoid passing this other parameter? [5 Marks]

iii: Consider the declaration

```
double a[10] = {1.2, 2.1, 3.3, 3.5, 4.5, 7.9, 5.4, 8.7, 9.9, 1.0};
```

Write a function named **out_of_order** that will test this array for the condition

```
a[0] <= a[1] <= a[2] <= ...
```

The function returns a -1 if the elements are **not** out of order, otherwise it returns the index of the first element that is out of order. Explain what you do to avoid out of bounds array access.   [10 marks]

Question 3

a) Give a general outline of a successful recursive function definition   .   **[3 marks]**

b) Why might a recursive solution to a problem run slower than an iterative version that does the same thing?   **[2 marks]**

c) Iterative solutions are always possible. Why then, would we bother with recursive solutions to problems? Give advantages that some recursive algorithms have over the iterative versions in your explanation.   **[3 marks]**

d) Write a recursive void function that has one parameter which is a positive integer. When called, the function is to write its arguments to the screen backward: If the argument is 1234, the output should be. 4321.   **[6 marks]**

e) Write a recursive version of the iterative function below: **[6 marks]**

```
int g(int n)
{
  int h = 1;
  while (n > 1)
  {
    h = h * n;
    n--;
  }
  return h;
}
```

Question 4
The following is a definition of a structure that represents a bank account.
```
struct Account
{
char fName[13], lName[13];
float balance;
int idNum;
};
```
and the following a declaration of an array of the structures.
```
Account accounts[13];
```
Use them to answer the questions that follow
a) Determine the size of one element of the array given that the size of the *char* data type is
1byte while the size of *int* and *float* is 4 bytes each. [5 Marks]
b) Determine the address of the SEVENTH element of the array if the address of the FIRST
element is 4002 (use the information in a) above. [5 Marks]
c) Write the statement that would initialize the *fName* of the seventh element to the string
"raia mwema". [5 Marks]
d) Write a piece of code that displays the sum of all the balances in the array. [5 Marks]

Question 5
a) Declare and open input file stream fileIn and output file stream fileOut. Attach these to files named
input.dat and output.dat. Write #include directives for any required header files. Give a brief
explanation for each statement you. **[7 marks]**
b) Files used in file I/O have two names. Give the names and explain how they are used.
[3 marks]
c) Assume you have opened and connected stream variables fileIn and fileOut in part (a) above.
Assume further that you have finished with the input and output files. Write the statements
necessary to close these files. **[2 marks]**
d) Assume that your program opens a file stream, has a file connected, and writes to the file. What
changes need to be made to make your program write to the screen? **[2 marks]**
e) You are writing a program. Give the necessary statements to open a file and to confirm that the file
has been successfully opened for writing. Why is it important to bother to test if the file has been
successfully opened? **[6 marks]**