2nd SEMESTER UNIVERSITY EXAMINATIONS – 2008/2009

IS 136: C Programming Language

Duration: 2 Hours

Date: 29th July, 2009

Instructions:

1. The examination is divided into Sections A and B.
2. Answer all questions in Section A and any three in Section B.
3. Section A carries 24 Marks and Section B carries 36 marks, giving a total of 60 marks.
4. This paper consists of 2 printed pages.

SECTION A

1. a. Describe the term function as used in C programming. (1.5 Marks)

   b. Write brief notes regarding C functions for each of the following. (4 Marks)
      i. Number of arguments taken and their ordering by data type
      ii. Number of arguments returned
      iii. Changes to values of arguments upon function return
      iv. Status of variables declared within a function upon function return

2. a. Explain the terms prototype and scope as used in C programming. (2 Marks)

   b. Explain what the following code snippet does. Justify your answer. (3 Marks)

```
#include <stdio.h>
void print_stars(int);
int i;

int main(){
        for (i=0; i < 5; i++)  print_stars(5);
        return 0;
}

void print_stars(int n){
        printf("*");
}
```

3. a. The C Language supports the constructs **enum**, **#define** and **const**. Use examples to illustr the syntax for each and state the relative difference between the three constructs. (3 Mark

   b. Distinguish between pass by value and pass by reference. What is the significance of the latter to the **scanf** function? (2 Marks)

4. a. Explain the term pointer and use examples to illustrate the meaning of the operators & and * in relation to pointers. (2 Marks)

   b. Write a function prototype for the function named '**change_argument**' with one argumen changed with the function. Use argument type of your own choice and assume the functio return any value. (3 Marks)

5. a. Explain each of the special streams **stdin**, **stdout**, and **stderr** as defined in the **stdio.h** hea (1.5 Mark

   b. Explain what each of the following code fragment does. (2 Marks)
      i. fprintf (stdout,"Hello World!\n");
      ii. char tLine[50]; fgets (tLine,50,stdin);