University of Ruhuna - Faculty of Technology

Bachelor of Information & Communication Technology Degree
Level 1 (Semester 1) Examination, October 2018

Course Unit: ICT1133- Fundamentals of Computer Programming (Theory Examination)

Answer all four (04) questions **Time Allowed: 2 hours**

IMPORTANT INSTRUCTIONS

- The medium of this examination is **English**
- This paper contains four (04) questions on five (05) pages.
- This is a **closed book** examination.

(1)

a) Briefly state what is a computer program.

b) Answer the followings related to Algorithms.

i) List down **two** advantages of developing an Algorithm in problem solving phase.

ii) Draw a flowchart which finds and prints the **Sum** of **Even** Numbers from **1 to N**. Take the value N as user input.

c) Briefly explain the following programming tasks with respect to C programming language using suitable examples.

- Variable declaration
- Variable initialization

d) Write **two** differences between keywords and identifiers in C language and give examples for **four different** types of **invalid** identifiers.

e) Write a single C statement to find the length of an **integer array** using the **sizeof()** operator.

f) What is Typecasting in C language? Give **one** example.

[20 marks]

(2)

a) What is a mixed mode arithmetic expression? Give **one** example.

b) Give the outputs for following C expressions if "a" is an integer.

i) a = 32 + 'E' / 4 (consider the ASCII value of 'A' as 65)

ii) a = 78 / 2 + 5 + 5*((10 - 2) % 2)

iii) a = !(x<=5) (consider x as 4)

iv) a = --x – y-- + ++z; (consider x = 5, y = 3, z = 4)

c) Find the **errors** in the following C code segments. Rewrite the corrected code.

i)
```
if( 5 <= x < 20)
    a = 5;
    b = 7;
else ( x >= 20)
    a = 20;
```

ii)
```
SWITCH( a ){
    case =15:
        printf("a is15");
        break;
    default :
        printf(a is not 15 or 16);
    case =16:
        printf("a is 16");
        break;
}
```

d) "break" and "continue" statements can be used inside a loop of a C program. Briefly explain the difference between "break" and "continue" statements with the aid of suitable C Programs. (Use **if** and **for loop** in your examples)

e) Convert the following for loop into a while loop in C. Write down the output of the following loop.

```
for( int i=15 ; i>=-5 ; i-=5 ){
        printf("%d ", i);
}
```

f) Clearly state what is wrong with the following while repetition statement, which is supposed to calculate the sum of the integers from 100 down to 1. Write the corrected code.

```
int z =100;
while ( z >= 0 )
        sum += z;
```

[25 marks]

(3)

a) Write **two** main properties of an array in C language.

b) In C, there is a way to pass command line arguments to a program when it begins execution. Answer the following questions based on command line arguments.

i) State the usage of two arguments passed into the main function when taking command line arguments.

```
void main ( int argc, char *argv[] )
```

ii) Write a function called "findEldest" to find the age of the eldest student in a classroom of 10 students. This function should take an array of student ages called **age[10]** as the **parameter** and **print** the age of the eldest student.

iii) Write a C main function to take age of ten(10) students as command line arguments and print the age of the eldest student. **Follow the instructions** given bellow.

1. Declare your main() to take age values as command line arguments.
2. Implement a validation with a suitable error message to have exactly 10 age values.
3. Convert the age values into integers and store them in an integer array called age[10].
4. Call the findEldest() function and pass the age array as the argument.

c) What is a recursive function?

d) List down **two** important factors you have to consider when passing arguments into function parameters (Mapping arguments into parameters).

e) Answer the below questions based on static variables.

i) Write down a special characteristic of a static variable.

ii) What are the outputs of the following programs A and B?

```
#include<stdio.h>
int update();

void main(){
   int x= 7;
   printf("value of a: %d\n",update(x));
   printf("value of a: %d\n",update(x));

}
int update(int y){
   int a=5;
   a+=y;
   return a;
}
```

Program A

```
#include<stdio.h>
int update();

void main(){
   int x= 7;
   printf("value of a: %d\n",update(x));
   printf("value of a: %d\n",update(x));

}
int update(int y){
   static int a=5;
   a+=y;
   return a;
}
```

Program B

f) Fill the blank lines (A, B, C, D and E) of the following program. The expected output of the program is given below.

```
#include<stdio.h>
----------------------------        //Line A. Include the appropriate library.

typedef struct{
        char name[20];
        float price;
}fruit;

void main(){
        ----------------------------        //Line B. Declare the structure variable
        char part1[10] = "Wood";
        char part2[10] = "Apple";
        ----------------------------        //Line C. Combine part1 & part2
        ----------------------------        //Line D. Copy the Combined string into structure member
                                                    called "name"
        fr.price = 50.00;

        printf("Fruit Name is: %s\n", fr.name);
        printf("Fruit has  %d characters \n",----------);      // Line E. Find the length of
                                                                    the name
}
```

Output :

```
Fruit Name is: WoodApple
Fruit has 9 characters
```

[30 marks]

(4)

a) What is the main difference between a simple variable and a pointer?

b) List two advantages of using dynamic memory allocation. Give suitable inbuilt C function names to explain each of the advantage.

c) Briefly discuss the main differences between **"r", "w"** and **"a"** modes in file opening. Consider the file is already exists and has text.

d) Examine the below program which is used to find and print the word count and vowel count of a given file. Fill the blank lines (A, B, C, D, E, F and G) by selecting the correct code segments from the given pool of C Codes.

```
1 #include<stdio.h>

2 void main(){

3       ----------------------------;                    //line A
4       int wordcount=1, vowelcount=0;
5       char ch;

6       read = ----------------------------;              //line B

7       if( ------------- ){                              //line C
8               printf("error opening the file");
9               return -1;
10      }
11      else{
12              while(-------------------){               //line D
13                      if( ---------)                    //line E
14                              wordcount++  ;
15                      else if(ch== 'a' || ch =='e' ||ch =='i' ||ch =='o' ||ch =='u' )
16                              ----------------------;   //line F
17              }
18      }
19      ----------------------;                           //line G

20      printf("\nNo of words in the file : %d\n",wordcount);
21      printf("\nNo of lowercase vowels in the file : %d\n",vowelcount);
22 }
```

Code Pool

[fopen("file.txt","w") , ch ==' ' , fclose(read) , ch == '\n' , read==0 , vowelcount++ , FILE *read , read==NULL , ch++ , fopen("file.txt","r") , (ch=getc(read)) !=EOF , (ch=putc(read)) !=EOF]

e) It is said that "Local variables inside a function will create at the beginning of the function execution and destroy when function terminates". Within that context use the following program to answer the questions.

```
#include<stdio.h>
int  b = 5;

void test(){
        //Line#1
        b++;
}
void main(){
        int a=2;
        printf("value of b: %d\n",b);
        test();
        printf("value of b: %d\n",b);
}
```

i) Identify local and global variables.

ii) What is the output of the program?

iii) Will there be any change in the output if you put **"int b=1;"** at //Line#1? If yes write the new output.

[25 marks]

---------- End of Paper ----------