Department of ICT
Faculty of Technology
University of Ruhuna

Programming Practicum – ICT1142 **Level 1- Semester 1**

Lab Sheet 10 **| 2022**

Objective:
The purpose of this lab session is to study basics on Pointers and String handling functions in C.

String Handling Functions

Exercise 1

Type and execute the following program and get the idea about string functions in C.

```
#include <stdio.h>
#include <string.h>

int main () {
   char str1[12] = "Hello";
   char str2[12] = "World";
   char str3[12];
   int  len ;

   strcpy(str3, str1);
   printf("strcpy( str3, str1) :  %s\n", str3 );

   strcat( str1, str2);
   printf("strcat( str1, str2):   %s\n", str1 );

   len = strlen(str1);
   printf("strlen(str1) :  %d\n", len );

   return 0;
}
```

Exercise 2

Type and run the following program.

```c
#include<stdio.h>
#include<string.h>

void main(){
    char s1[20], s2[20], s3[20];
    int x, l1,l2,l3;

    printf("enter two strings:");
    scanf("%s %s",s1,s2);

    x = strcmp(s1,s2);

    if(x!=0){
        printf("strings are not equal\n");
        strcat(s1,s2);
    }
    else
        printf("strings are equal\n");

        strcpy(s3,s1);
        l1 = strlen(s1);
        l2 = strlen(s2);
        l3 = strlen(s3);

        printf("s1=%s length=%d characters\n",s1,l1);
        printf("s2=%s length=%d characters\n",s2,l2);
        printf("s3=%s length=%d characters\n",s3,l3);
}
```

Exercise 3

Type and run the following program to get an idea about gets() and puts() functions.

```c
#include<stdio.h>
int main( ){
   char name[30];
   printf("Enter name: ");
   gets(name);     //Function to read string from user.
   printf("Name: ");
   puts(name);    //Function to display string.
   return 0;
}
```

Pointers

How to define a pointer

```
type *name;

int    *ip;  /* pointer to an integer */
double *dp;  /* pointer to a double */
float  *fp;  /* pointer to a float */
char   *ch   /* pointer to a character */
```

Exercise 4

Type and execute the following program and get an idea about pointer basics.

```
int main() {
        int k;
        int *ptr;
        k=5;
        ptr=&k;

        printf("value of k = %d and it is stored at %p\n", k, &k);
        printf("value of ptr = %p and it is stored at %p\n", ptr, &ptr);
        printf("the value of the integer pointed to by ptr is = %d \n", *ptr);
        return 0;
}
```

Exercise 5

Write two functions to interchange two integers. One function should pass the parameters by reference (pchange()) while the other one should pass the parameters by values (change()). Call the function from main function and check the answers.

```
void change(int x, int y);
void pchange(int *a, int *b);
```

Exercise 6

Consider the following code sample that demonstrates the Pass by Reference. Analyze the answers.

```
#include <stdio.h>
void twice (int * val);

int main()
{       /*
                & = "Address of …"
                * = "Content of …"
        */
  int x=56;
  int *y;
  y=&x;
  twice(y);
  printf("y  value= %p\n",y);
  printf("value referred by y = %d\n",*y);
  printf("x value=%d\n",x);
  return 0;
}

void twice (int *val){
  *val=(*val)*2;
}
```

Exercise 7

i) Write a C program to add two numbers using pointers.

ii) Modify the same program by writing a function for this task. (Pass two numbers as Pointers / Pass By Reference)

Exercise 8

Write a C program according to the following guidelines.

• Declare integer, float and character variable called num, fl and ch respectively.
• Declare integer, float and character pointer variables called numptr, flptr and chptr.
• Assign values 154, 78.5 and ‘g’to variables num, fl and ch respectively.
• Assign address of num to numptr.
• Assign address of fl to flptr.
• Assign address of ch to chptr.
• Print the value of *numptr, numptr and &numptr
• Print the value of *flptr, flptr and &flptr.
• Print the value of *chptr, chptr and &chptr.
• Now add (*chptr)++; to your program and print the value of ch
• Do this to other two pointer variables and print the value of num and fl.