Assignment - 20 A Job Ready Bootcamp in C++, DSA and IOT MySirG

Pointers

1. Write a function to swap values of two in variables of calling function. (TSRS)

Program -

```
#include<stdio.h>

int swap(int*,int*);
int main()
{
  int a , b , c;
  printf("Enter value of a : ");
  scanf("%d",&a);
  printf("Enter value of b : ");
  scanf("%d",&b);
  c = a;
  a = swap(&a , &b);
  b = swap(&b , &c);

  printf("After swapping :-\n");
  printf("Value of a = %d\nValue of b = %d",a,b);
  return 0;
}

int swap(int *x , int *y)
{
   *x = *y;
   return *x;
}
```

Output -

Enter value of a : 2
Enter value of b : 10
After swapping :-
Value of a = 10
Value of b = 2

---

2. Write a function to swap strings of two char arrays of calling functions. (TSRS)

Program -

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

char *swapString(char *, char *);

int main()
{
```

```
  char str1[50], str2[50], str3[50];

  printf("Enter string number - 1 : ");
  fgets(str1, 50, stdin);
  printf("Enter string number - 2 : ");
  fgets(str2, 50, stdin);

  strcpy(str3, str1);
  strcpy(str1, swapString(str1, str2));
  strcpy(str2, swapString(str2, str3));

  printf("\nAfter swapping :-\n");
  printf("String - 1 : %s", str1);
  printf("String - 2 : %s", str2);

  return 0;
}

char *swapString(char *ptr1, char *ptr2)
{
  int i;
  for (i = 0; ptr2[i]; i++)
  {
    ptr1[i] = ptr2[i];
  }
  return ptr1;
}
```

Output -
Enter string number - 1 : andrew james
Enter string number - 2 : anmol zakie

After swapping :-
String - 1 : anmol zakie

String - 2 : andrew james

---

3. Write a function to sort an array of int type values. [void sort(int *ptr,int size);]

Program -

```
#include <stdio.h>

void sort(int*,int);

int main() {
  int arr[5] , i;

  printf("Enter 5 array values\n");
  for(i = 0 ; i < 5 ; i++)
    scanf("%d",&arr[i]);
```

```
  sort(arr,5);
  printf("Sorted array is:-\n");
  for(i = 0 ; i < 5 ; i++)
  {
    printf("%d ",arr[i]);
  }
  return 0;
}

void sort(int *ptr,int size)
{
  int i , j;
  for(i = 0 ; i < size ; i++)
    {
      for(j = i+1 ; j < size ; j++)
        {
          if(*(ptr + i) > *(ptr + j))
          {
            *(ptr + i) = *(ptr + i) + *(ptr + j);
            *(ptr + j) = *(ptr + i) - *(ptr + j);
            *(ptr + i) = *(ptr + i) - *(ptr + j);
          }
        }
    }
}
```

Output -
Enter 5 array values
0 -1 4 2 8
Sorted array is:-
-1 0 2 4 8

---

4. Write a program in C to demonstrate how to handle the pointers in the program.

Program -

```
#include<stdio.h>

int main()
{
  int x = 3, y = 5 , *ptr1 = NULL, *ptr2 = NULL;

  ptr1 = &x , ptr2 = &y;
  if(ptr1 != NULL)
  {
     *ptr1 = 9;
  }
  if(ptr2 != NULL)
  {
    *ptr2 = 8;
  }
```

```
  printf("x = %d , y = %d",x,y);
  return 0;
}
```

Output -
x = 9 , y = 8

---

5. Write a program to find the maximum number between two numbers using a pointer

Program -
```
#include<stdio.h>

int main()
{
  int a , b , *ptr;

  printf("Enter two numbers : ");
  scanf("%d%d",&a,&b);

  ptr = a > b ? &a : &b;

  printf("Maximum number is : %d",*ptr);
  return 0;
}
```

Output -
Enter two numbers : 9 7
Maximum number is : 9

---

6. Write a program to calculate the length of the string using a pointer

Program -
```
#include<stdio.h>

int length(char *);
int main()
{
  char str[50];

  printf("Enter a string : ");
  fgets(str,50,stdin);

  printf("Length of the string is : %d",length(str));
  return 0;
}

int length(char *ptr)
```

```
{
  int i;
  for(i = 0 ; *(ptr + i) ; i++);
  return i-1;
}
```

Output -

Enter a string : Bhopal Madhya Pradesh
Length of the string is : 21

---

7. Write a program to count the number of vowels and consonants in a string using a pointer.

Program -

```
#include <stdio.h>

int countVowels(char *);
int countConsonants(char *);

int main()
{
  char str[50];

  printf("Enter a string : ");
  fgets(str, 50, stdin);

  printf("Number of vowels : %d", countVowels(str));
  printf("\nNumber of consonants : %d", countConsonants(str));
  return 0;
}

int countVowels(char *ptr)
{
  int i, count = 0;
  for (i = 0; ptr[i]; i++)
  {
    if (ptr[i] == 'a' || ptr[i] == 'e' || ptr[i] == 'i' || ptr[i] ==
'o' || ptr[i] == 'u')
      count++;
    if (ptr[i] == 'A' || ptr[i] == 'E' || ptr[i] == 'I' || ptr[i] ==
'O' || ptr[i] == 'U')
      count++;
  }
  return count;
}

int countConsonants(char *ptr)
{
  int i , count = 0;

  for(i = 0 ; ptr[i] ; i++)
```

```
  {
    if((ptr[i] >= 'a' && ptr[i] <= 'z') || (ptr[i] >= 'A' && ptr[i]
       <= 'Z'))
    {
      if (ptr[i] == 'a' || ptr[i] == 'e' || ptr[i] == 'i' || ptr[i]
           == 'o' || ptr[i] == 'u')
      continue;
    else if (ptr[i] == 'A' || ptr[i] == 'E' || ptr[i] == 'I' ||
          ptr[i] == 'O' || ptr[i] == 'U')
      continue;

      count++;
    }
  }
  return count;
}
```

Output -
Enter a string : Hello world!
Number of vowels : 3
Number of consonants : 7

---

8. Write a program to compute the sum of all elements in an array using pointers.

Program -

```
#include<stdio.h>

int computeSum(int * , int);
int main()
{
  int arr[10] , i;

  printf("Enter 10 array elements\n");
  for(i = 0 ; i < 10 ; i++)
    scanf("%d",&arr[i]);

  printf("Sum of all the array elements : %d",computeSum(arr,10));
  return 0;
}

int computeSum(int *ptr , int l)
{
  int i , sum = 0;
  for(i = 0 ; i < l ; i++)
  {
    sum += ptr[i];
  }
  return sum;
}
```

Output -
Enter 10 array elements
1 0 2 10 12 20 8 2 2 6
Sum of all the array elements : 63

---

9. Write a program to print the elements of an array in reverse order.

Program -

```
#include <stdio.h>

void reverseArray(int *, int);
int main()
{
  int arr[10], i;

  printf("Enter 10 array elements\n");
  for (i = 0; i < 10; i++)
    scanf("%d", &arr[i]);

  printf("Array in reverse order:-\n");
  reverseArray(arr, 10);

  return 0;
}

void reverseArray(int *ptr, int l)
{
  int i;
  for (i = l - 1; i >= 0; i--) {
    printf("%d ", ptr[i]);
  }
}
```

Output -
Enter 10 array elements
1 2 3 4 5 6 7 8 9 10
Array in reverse order:-
10 9 8 7 6 5 4 3 2 1

---

10. Write a program to print a string in reverse using a pointer

Program -

```
#include <stdio.h>

void reverseString(char*,int);
int main()
{
  char str[50];
  int l;
  printf("Enter a string : ");
  fgets(str,50,stdin);

  for(l = 0 ; str[l] ; l++);

  reverseString(str,l-1);

  printf("String in reverse : %s",str);
  return 0;
}

void reverseString(char *ptr , int len)
{
  int i;
  char s;
  for(i = 0 ; i < len/2 ; i++)
  {
      s = ptr[i];
      ptr[i] = ptr[len-1-i];
      ptr[len-1-i] = s;
  }
}
```

Output -

Enter a string : Shahrukh khan
String in reverse : nahk hkurhahS
