

Raunak Gayen
2020EEB066

1. Write a C program to add two numbers using pointers.

```
1 #include<stdio.h>
2 void sum(int *A, int *B)
3 {
4     int res;
5     res = *A + *B;
6     printf("The sum is = %d",res);
7 }
8 int main()
9 {
10    int a,b;
11    printf("Enter 2 numbers\n");
12    scanf("%d%d",&a,&b);
13    sum(&a,&b);
14    return 0;
15 }
```

```
D:\Assignment>a.exe
Enter 2 numbers
4 5
The sum is = 9
```

2. Write a C program to swap two numbers using pointers.

```
1 #include<stdio.h>
2 void swap(int *a, int *b)
3 {
4     int temp;
5     temp=*a;
6     *a=*b;
7     *b=temp;
8     printf("after swapping a=%d and b=%d",*a,*b);
9 }
10 int main()
11 {
12     int a,b;
13     printf("enter 2 numbers\n");
14     scanf("%d%d",&a,&b);
15     printf("before swapping, a=%d and b=%d\n\n",a,b);
16     swap(&a,&b);
17     return 0;
18 }
```

```
D:\Assignment>a.exe
enter 2 numbers
6 12
before swapping, a=6 and b=12
after swapping a=12 and b=6
```

3. Write a C program to sort an array of ten integer values in ascending order. Your function header for sorting should be `sortvalue(int *p, int n)`.

```
1 #include<stdio.h>
2 void sort(int *arr, int n)
3 {
4     for(int i=0;i<n;i++)
5     {
6         for(int j=i+1;j<n;j++)
7         {
8             if(*(arr+i) > *(arr+j))
9             {
10                 int temp;
11                 temp = *(arr+i);
12                 *(arr+i) = *(arr+j);
13                 *(arr+j)=temp;
14             }
15         }
16     }
17     //printing the final result
18     printf("The final array after sorting --->\n\n");
19     for(int i = 0; i < n; i++)
20     {
21         printf("%d ",*(arr+i));
22     }
23
24 }
25 int main()
26 {
27     int n;
28     printf("enter array size\n");
29     scanf("%d",&n);
30     int arr[n];
31     printf("enter the array elements one by one\n");
32     for (int i = 0; i < n; i++)
33     {
34         scanf("%d",&arr[i]);
35     }
36     sort(arr,n);
37     return 0;
38 }
```

```
D:\Assignment>a.exe
enter array size
6
enter the array elements one by one
9 5 2 1 4 7
The final array after sorting --->
1 2 4 5 7 9
```

4. Write a C program to accept five integer values from keyboard using pointer. Write a function to print the elements of the array in reverse order.

```
1 #include<stdio.h>
2 void reverse(int *arr,int n)
3 {
4     printf("enter the array elements\n");
5     for (int i = 0; i < n; i++)
6     {
7         scanf("%d", (arr+i));
8     }
9     for(int i=n-1;i>=0;i--)
10    {
11        printf("%d ", *(arr+i));
12    }
13
14 }
15 int main()
16 {
17     int n=5;
18     int arr[n];
19     //function for taking input and
20     //reversal
21     reverse(arr,n);
22     return 0;
23 }
```

```
D:\Assignment>a.exe
enter the array elements
2 3 1 4 5
5 4 1 3 2
```

5. Suppose the following strings are to be sorted (containing a maximum of 12 characters): PACIFIC, ATLANTIC, INDIAN, CARIBBEAN, BLACK, RED, NORTH, BALTIC, CASPIAN. Write a C program to sort the above strings into alphabetical order using an array of pointers.

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<string.h>
4 void sort(char *str[],int n)
5 {
6     char temp[20];
7     for(int i=0;i<n;i++)
8     {
9         for(int j=i+1;j<n;j++)
10        {
11            if(strcmp(str[i],str[j])>0)
12            {
13                strcpy(temp,str[i]);
14                strcpy(str[i],str[j]);
15                strcpy(str[j],temp);
16            }
17        }
18    }
19    printf("after sorting ,the final list is-->\n\n");
20    for (int i = 0; i < n; i++)
21    {
22        printf("%s\n",str[i]);
23    }
24
25 }
26 int main()
27 {
28     int n;
29     printf("enter number of strings\n");
30     scanf("%d",&n);
31     char *str[20];
32     for (int i = 0; i < n; i++)
33     {
34         printf("Enter string %d\n",i+1);
35         str[i] = (char *)malloc(20*sizeof(char));
36         scanf("%s",str[i]);
37     }
38     sort(str,n);
39
40     return 0;
41 }
```

5. Suppose the following strings are to be sorted (containing a maximum of 12 characters): PACIFIC, ATLANTIC, INDIAN, CARIBBEAN, BLACK, RED, NORTH, BALTIC, CASPIAN. Write a C program to sort the above strings into alphabetical order using an array of pointers.

OUTPUT:

```
D:\Assignment>a.exe
enter number of strings
9
Enter string 1
PACIFIC
Enter string 2
ATLANTIC
Enter string 3
INDIAN
Enter string 4
CARIBBEAN
Enter string 5
BLACK
Enter string 6
RED
Enter string 7
NORTH
Enter string 8
BALTIC
Enter string 9
CASPIAN
after sorting ,the final list is-->

ATLANTIC
BALTIC
BLACK
CARIBBEAN
CASPIAN
INDIAN
NORTH
PACIFIC
RED
```