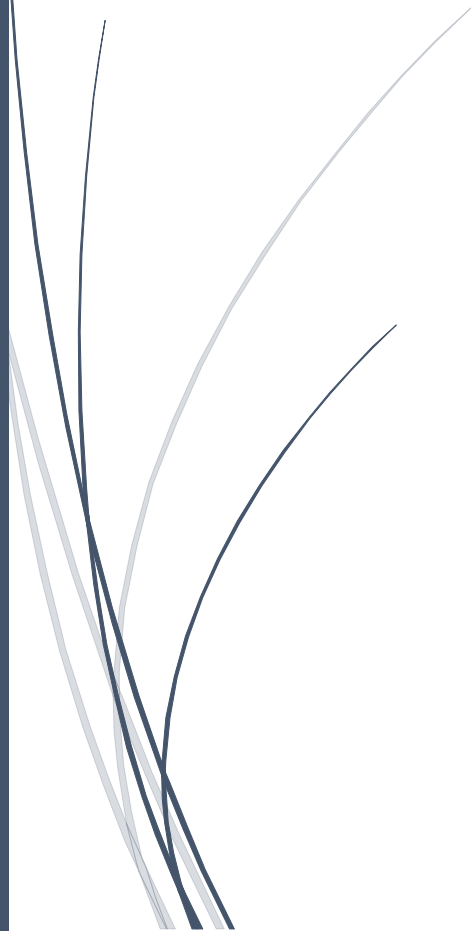
11/25/2022

PF Lab Assignment

Lab 11

Abdullah Shafiq
22K-4489



Task 01

Write a program to find out the greatest and the smallest among three numbers using pointers.

PROGRAM & OUTPUT

```c
#include<stdio.h>

int main()
{
    int a,b,c,*pa, *pb, *pc;

    printf("Enter three numbers:\n");
    scanf("%d%d%d", &a,&b,&c);
    pa= &a;
    pb= &b;
    pc= &c;
    if(*pa > *pb && *pa > *pc)
    {
        printf("Largest is: %d", *pa);
    }
    else if(*pb > *pc && *pb > *pc)
    {
        printf("Largest is : %d", *pb);
    }
    else if(*pa < *pb && *pa < *pc)
    {
        printf("Smallest is: %d", *pa);
    }
    else if(*pb < *pc && *pb < *pc)
    {
        printf("Smallest is : %d", *pb);
    }
    else
    {
        printf("Largest = %d", *pc);
    }
    return 0;
}
```

```
Enter three numbers:
34 56 78
Smallest is: 34
--------------------------
```

Task 02

Write a C program to swap corresponding elements of two arrays using pointers.

PROGRAM & OUTPUT

```c
#include <stdio.h>
void swap(int *ptr1, int *ptr2);
void main()
{
    int x[10];
    int j, n;
    printf("How many Elements...\n");
    scanf("%d", &n);
    printf("Enter Elements one by one\n");
    for (j = 0; j < n; j++)
    {
        scanf("%d", x + j);
    }
    swap(x + 0, x + 1);
    printf("\nResultant Array...\n");
    for (j = 0; j <n ; j++)
    {
        printf("X[%d] = %d\n", j, x[j]);
    }
}
void swap34(int *ptr1, int *ptr2 )
{
    int  temp;
    temp = *ptr1;
    *ptr1 = *ptr2;
    *ptr2 = temp;
}
```

```
How many Elements...
2
Enter Elements one by one
34
56

Resultant Array...
X[0] = 56
X[1] = 34

----------------------------
```

Task 03

Write a program that implements the function(WordCount).
int WordCount(char *Text, int *size);

PROGRAM & OUTPUT

```c
#include<stdio.h>
#include<string.h>
void wordcount(char a[],int d)
{
    int i;
    int count=0;
    char *ptr[20];
    for (i=0;i<d;i++)
    {
        ptr[i]=&a[i];
    }
    for (i=0;i<d;i++)
    {
        if(a[i]==' ')
        {
            count++;
        }
        else if (a[i]=='\0')
        {
            count++;
        }
    }
    printf("%d",count);
}
int main()
{
    char a[20];
    fgets(a,sizeof(a),stdin);
    int d=strlen(a);
    wordcount(&a,d);
}
```

```
Apple is Good
3
--------------------
```

Task 04

Write a C program to add two matrices using pointers. Create a function called calMat() that take pointers of 2 matrices as arguments and return the resulted sum and display it in main.

PROGRAM & OUTPUT

```c
#include<stdio.h>
#include<string.h>
void calmat()
{
    int a[2][2],b[2][2],c[2][2];
    int i,j;
    int *ptr1[2][2],*ptr2[2][2],*ptr3[2][2];
    printf("Enter a first matrix\n");
    for (i=0;i<2;i++)
    {
        for (j=0;j<2;j++)
        {
            printf("Enter a 1st Matrix in Pocket [%d][%d]",i,j);
            scanf("%d",&a[i][j]);
            ptr1[i][j]=&a[i][j];
        }
    }
    printf("Eenter a second matrix\n");
    for (i=0;i<2;i++)
    {
        for (j=0;j<2;j++)
        {
            printf("Enter a 2nd Matrix in Pocket [%d][%d]",i,j);
            scanf("%d",&b[i][j]);
            ptr2[i][j]=&b[i][j];
        }
    }
    printf("\nSum of Two matrices are:\n");
    for (i=0;i<2;i++)
    {
        for (j=0;j<2;j++)
        {
            c[i][j]=*ptr1[i][j]+*ptr2[i][j];
            ptr3[i][j]=&c[i][j];
        }
    }
    for (i=0;i<2;i++)
    {
        for (j=0;j<2;j++)
        {
            printf("%4d",*ptr3[i][j]);

        }
        printf("\n");
    }
}
int main()
{
    calmat();
}
```

```
Enter a first matrix
Enter a 1st Matrix in Pocket [0][0]1
Enter a 1st Matrix in Pocket [0][1]2
Enter a 1st Matrix in Pocket [1][0]3
Enter a 1st Matrix in Pocket [1][1]4
Eenter a second matrix
Enter a 2nd Matrix in Pocket [0][0]5
Enter a 2nd Matrix in Pocket [0][1]6
Enter a 2nd Matrix in Pocket [1][0]7
Enter a 2nd Matrix in Pocket [1][1]8

Sum of Two matrices are:
   6   8
  10  12

--------------------------------
```

Task 05

Write a function countEven(int*, int) which receives an integer array and its size, and returns the number of even numbers in the array.

PROGRAM & OUTPUT

```c
#include<stdio.h>
#include<string.h>
void even(int a[])
{
    int *ptr[10];
    int i;
    int count=0;
    for (i=0;i<10;i++)
    {
        ptr[i]=&a[i];
    }
    for (i=0;i<10;i++)
    {
        if(a[i]%2==0)
        {
            count++;
        }
    }
    printf("There are [%d] even Numbers in the Givne data.",count);
}
int main()
{
    int a[10]={2,5,7,8,3,6,10,34,67,56};
    even(&a);
}
```

```
There are [6] even Numbers in the Givne data.
--------------------------------
```

Task 07

Create a structure with student details and print the inputted details (Id, Name, Marks, Grade). Memory to store and print structure will be allocated at run time by using malloc().

PROGRAM & OUTPUT

```c
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
struct details
{
    int id;
    char name[20];
    int marks;
    int grade;
};
int main()
{
    struct details *ptr;
    int n;
    printf("Enter Number of records: ");
    scanf("%d",&n);
    ptr=(struct details *)malloc(n * sizeof(struct details));
    for (int i=0;i<n;++i)
    {
        printf("Enter Id, Name, Marks, Grade: ");
        scanf("%d %s %d %d",&(ptr+i)->id,&(ptr+i)->name,&(ptr+i)->marks,&(ptr+i)->grade);
    }
    printf("Displaying Detals:\n");
    for (int i=0;i<n;++i)
    {
        printf("ID: %d\nName: %s\nMarks: %d\nGrade:%d",(ptr+i)->id,(ptr+i)->name,(ptr+i)->marks,(ptr+i)->grade);
    }
    free(ptr);
}
```

```
Enter Number of records: 1
Enter Id, Name, Marks, Grade: 4489 Abdullah 89 1
Displaying Detals:
ID: 4489
Name: Abdullah
Marks: 89
Grade:1
--------------------------------
```

Task 06

Write a program that implements the SortFunction that takes argument pointer to an array, size of the array and the
order in which it is going to be sort. Such as, 1 for Asscending order and 2 for Descending orde. Finally, print this
array in Main() to check.
void SortFunction(int *arr, int *size, int order);

PROGRAM & OUTPUT

```c
#include<stdio.h>
#include<string.h>
void AscSorting(int *a, int d)
{
    int i=0,j=0,temp=0;
    int *b[5];
    for (i=0;i<5;i++)
    {
        b[i]=&a[i];
    }
    for (i=0;i<5;i++)
    {
        for (j=i+1;j<5;j++)
        {
            if(*b[i]>*b[j])
            {
                temp=*b[i];
                *b[i]=*b[j];
                *b[j]=temp;
            }
        }
    }
    for (i=0;i<5;i++)
    {
        printf("%d",*b[i]);
    }
}
void DscSorting(int *a, int d)
{
    int i=0,j=0,temp=0;
    int *b[5];
    for (i=0;i<5;i++)
    {
        b[i]=&a[i];
    }
    for (i=0;i<5;i++)
    {
        for (j=i+1;j<5;j++)
        {
            if(*b[i]<*b[j])
            {
            temp=*b[i];
            *b[i]=*b[j];
            *b[j]=temp;
            }
        }
    }
    for (i=0;i<5;i++)
    {
        printf("%d",*b[i]);
    }
}
```

```c
int main()
{
    int a[]={3,6,1,5,2};
    int option;
    int d=strlen(a);
    printf("Select:\n1.Ascending Sorting\n2.decsending Sorting\n");
    scanf("%d",&option);
    switch(option)
    {
    case 1:
        {
            AscSorting(a,d);
        }
        break;
    case 2:
        {
            DscSorting(a,d);
        }
    }
}
```

```
Select:
1.Ascending Sorting
2.decsending Sorting
1
12356
------------------------
```

```
Select:
1.Ascending Sorting
2.decsending Sorting
2
65321
------------------------
```