Lab assignment

NAME-Bibek kc ROLL NO.20051722 BRANCH -CSE

LAB -1

DATE- 07-26

1.

#include<stdio.h>

int main(){

int a,b,c,sum;

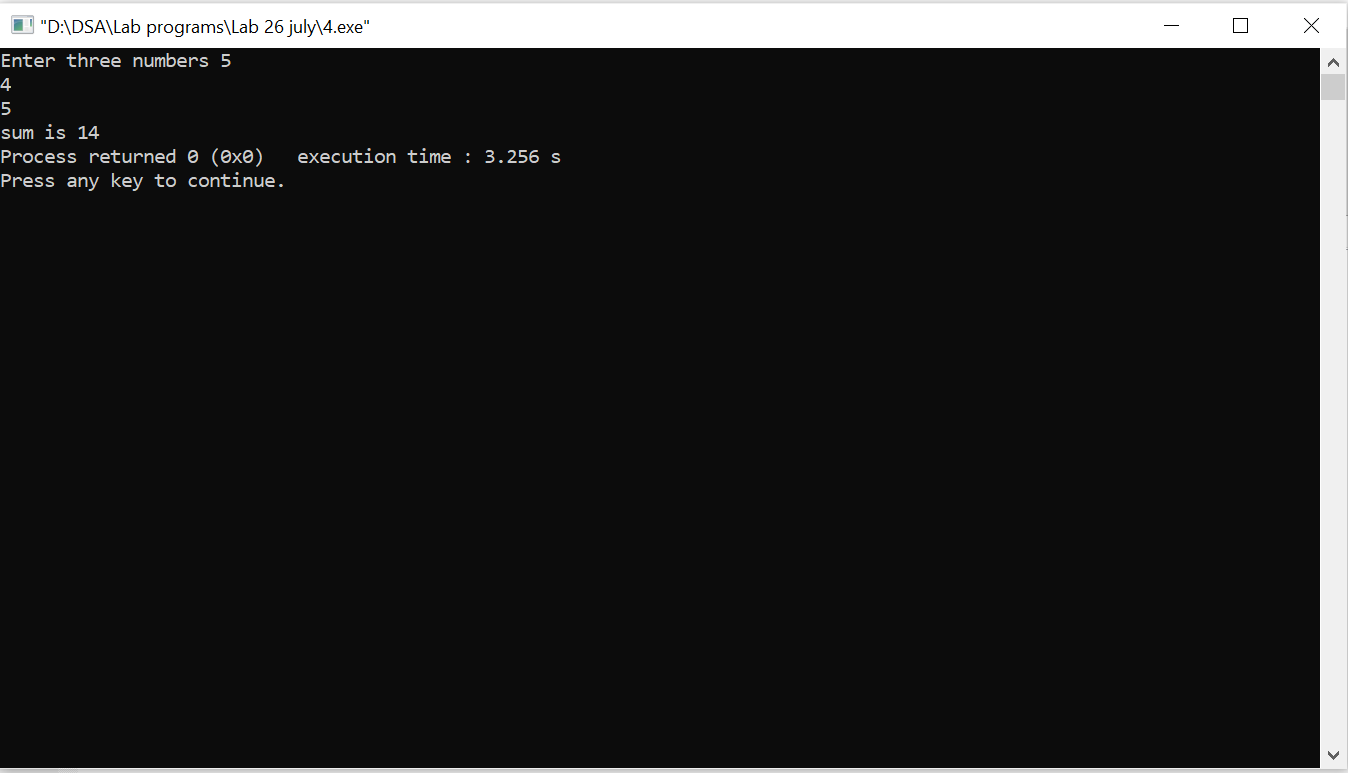
printf("Enter three numbers ");

scanf("%d%d%d",&a,&b,&c);

sum=a+b+c;

printf("sum is %d",sum);

}



2.

#include<stdio.h>

int main()

{

int a,b=1;

printf("Enter a number = ");

scanf("%d",&a);

while(a!=0){

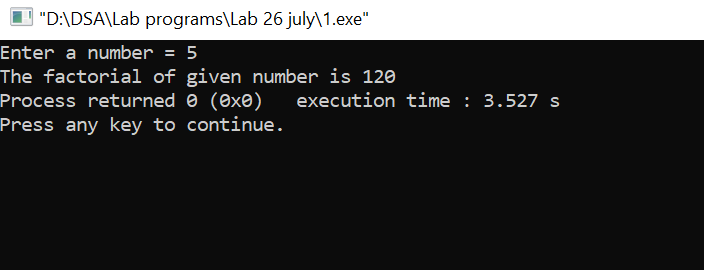
b\*=a;

a--;

}

printf("The factorial of given number is %d",b);

}



3.

#include<stdio.h>

int main()

{

int a;

printf("Enter a number\t");

scanf("%d",&a);

if(a%2==0)

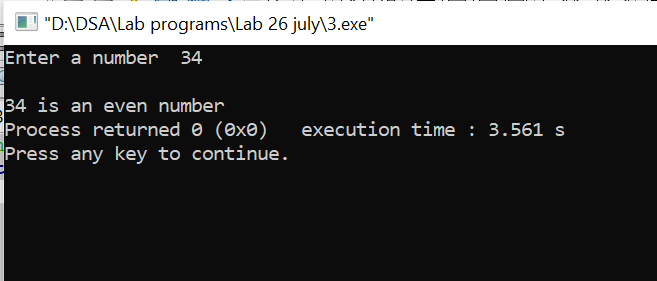
printf("\n%d is an even number",a);

else

printf("\n%d is an odd number",a);

return 0;

}



4.

#include<stdio.h>

int main(){

int a=5,sum=0;

while(a<=100){

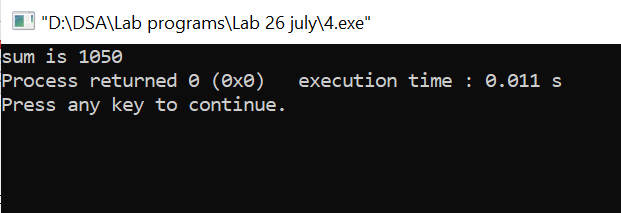
sum+=a;

a+=5;

}

printf("sum is %d",sum);

}



LAB-2

DATE- 08-02

1.

#include <stdio.h>

int main()

{

int n,s,i,a[100],b;

printf("Enter the size of an array=\t");

scanf("%d",&n);

for(i=0;i<n;i++)

scanf("%d",&a[i]);

b=a[0];

for(i=1;i<n;i++){

if(a[i]>b)

b=a[i];

}

printf("\n\nThe largest of array is = %d",b);

s=a[0];

for(i=1;i<n-1;i++){

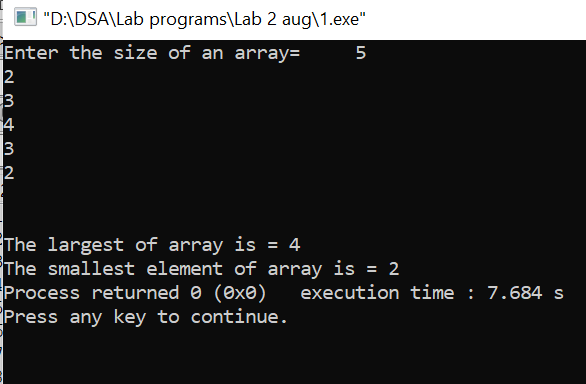
if(a[i]<s)

s=a[i];}

printf("\nThe smallest element of array is = %d",s);

return 0;

}



2.

#include<stdio.h>

#include<conio.h>

int main(){

int n,a,b,c[100],i;

printf("enter size of array:");

scanf("%d",&n);

printf("enter the highest and lowest range of array =");

scanf("%d%d",&a,&b);

printf("enter elements of array:\n");

for(i=0;i<n;i++){

scanf("%d",&c[i]);

}

printf("The elements between %d and %d are ",a,b);

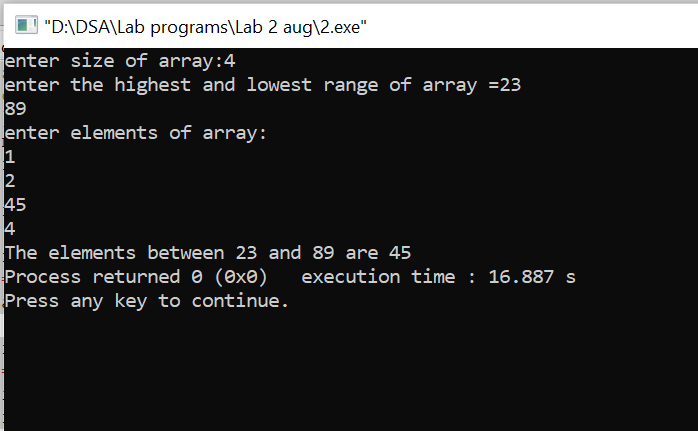
for(i=0;i<n;i++){

if(c[i]>a && c[i]<b)

printf("%d\t",c[i]);

}

}



3.

#include<stdio.h>

#include<conio.h>

main(){

int n,a[100],i;

printf("enter size of array:");

scanf("%d",&n);

printf("enter elements of array:\n");

for(i=0;i<n;i++){

scanf("%d",&a[i]);

}

printf("elements of array are:\n");

for(i=0;i<n;i++){

printf("%d\t",a[i]);

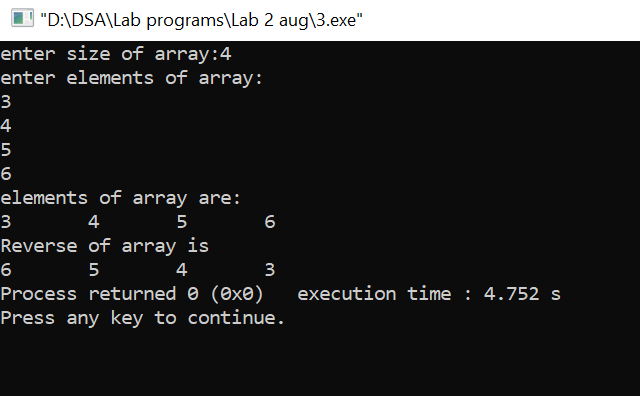
}

printf("\nReverse of array is\n");

for(i=n-1;i>=0;i--){

printf("%d\t",a[i]); }

return 0; }



4.

#include<stdio.h>

#include<conio.h>

int main(){

int n,c,b,a[100],i;

printf("enter size of array:");

scanf("%d",&n);

printf("enter a element to be searched=");

scanf("%d",&b);

printf("enter elements of array:\n");

for(i=0;i<n;i++){

scanf("%d",&a[i]);

}

for(i=0;i<n;i++){

if(b==a[i]){

c++;

}

}

if(c>=1)

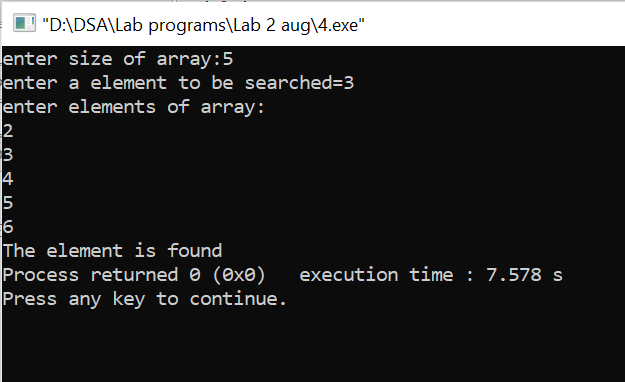
printf("The element is found");

else

printf("The element is not found");

return 0;

}



5.

#include<iostream>

#include<conio.h>

using namespace std;

struct mat{

int a[20][20];

int n;

void matrix(){

cout<<"Enter the numbers of rows or column";

cin>>n;

cout<<"Enter the elements of "<<n<<"\*"<<n<<" matrix"<<endl;

for(int i=0;i<n;i++)

for(int j=0;j<n;j++)

{

cin>>a[i][j];

}

cout<<"The elements of "<<n<<"\*"<<n<<" matrix is "<<endl;

for(int i=0;i<n;i++){

for(int j=0;j<n;j++)

{

cout<<a[i][j]<<"\t";

}

cout<<"\n";

}

}

void nonzero(){

int b=0;

for(int i=0;i<n;i++)

for(int j=0;j<n;j++){

if (a[i][j]==0)

b+=1;

}

cout<<"\nThe number of nonzero elements of given matrix is "<<b;

}

void ld(){

int b=0;

for(int i=0;i<n;i++)

for(int j=0;j<n;j++){

if (i==j)

b+=a[i][j];

}

cout<<"\nThe sum of the elements of the main diagonal is "<<b;

}

};

int main()

{

mat m;

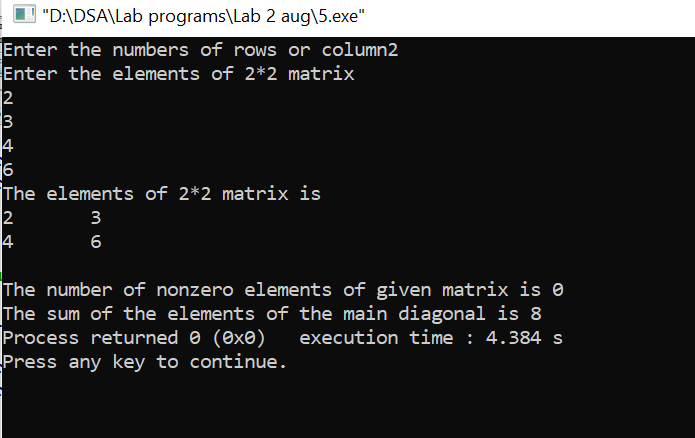
m.matrix();

m.nonzero();

m.ld();

return 0;

}



6.

#include<iostream>

using namespace std;

class employee{

double ebp;

double egp;

char ename[100];

char edisg[100];

char egender[10];

char edepart[20];

public:

void data()

{

int n;

cout<<"Enter number of employees\t";

cin>>n;

while(n!=0){

cout<<"Enter employee's name=\t";

cin>>ename;

cout<<"Enter employee's designation=\t";

cin>>edisg;

cout<<"Enter employee's gender M/F=\t";

cin>>egender;

cout<<"Enter employee's department=\t";

cin>>edepart;

cout <<"Enter employee's basic pay=\t";

cin>>ebp;

egp=ebp+ebp/4+ebp\*3/4;

cout<<"The gross pay of employee is =\t"<<egp<<endl;

cout<<"\n";

n--;

}

};

};

int main()

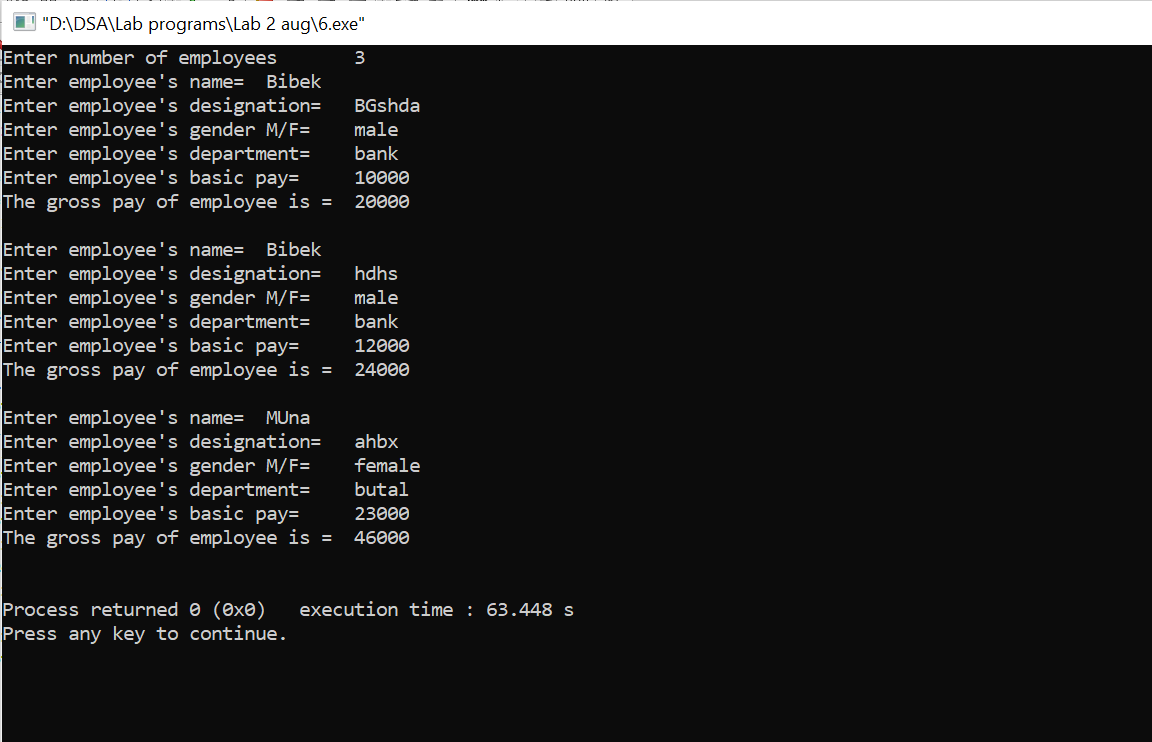
{

employee e;

e.data();

return 0;

}



LAB-3

DATE-08-09

1.

#include<iostream>

using namespace std;

class employee{

double ebp;

double egp;

char ename[100];

char edisg[100];

char egender[10];

char edepart[20];

public:

void data()

{

cout<<"Enter employee's name=\t";

cin>>ename;

cout<<"Enter employee's designation=\t";

cin>>edisg;

cout<<"Enter employee's gender M/F=\t";

cin>>egender;

cout<<"Enter employee's department=\t";

cin>>edepart;

cout <<"Enter employee's basic pay=\t";

cin>>ebp;

egp=ebp+ebp/4+ebp\*3/4;

cout<<"The gross pay of employee is =\t"<<egp<<endl;

cout<<"\n";

};

};

int main()

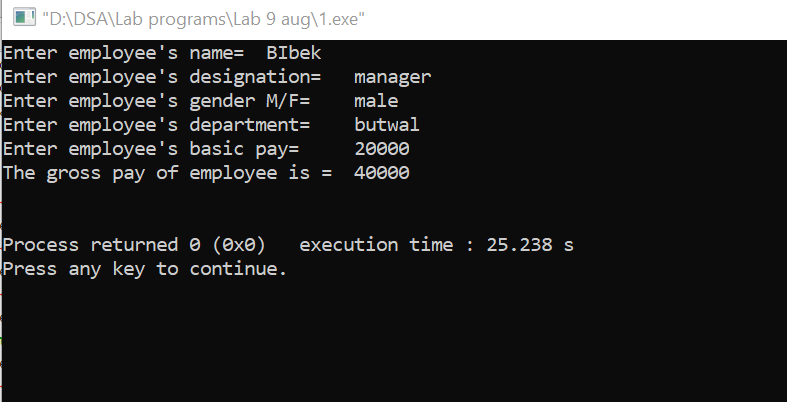
{

employee e;

e.data();

return 0;

}



2.

#include<stdio.h>

#include<stdlib.h>

int main()

{

int \*ptr,i;

ptr=(int\*)malloc(10\*sizeof(int));

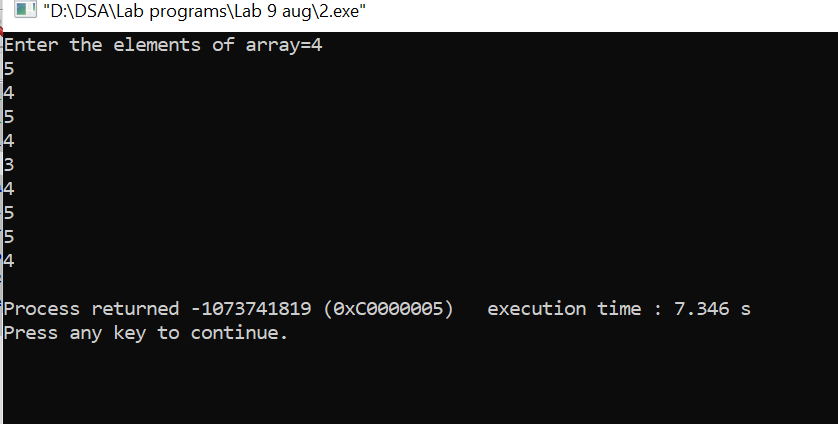
printf("Enter the elements of array=");

for(i=0;i<10;i++)

scanf("%d",&ptr+i);

return 0;

}



3.

#include<stdio.h>

#include<stdlib.h>

int main()

{

int \*ptr,n,i,x,flag=0;

printf("enter the length of the array");

scanf("%d",&n);

ptr=(int\*)malloc(n\*sizeof(int));

printf("enter the elements of the array");

for (i=0;i<n;i++)

scanf("%d",ptr+i);

printf("enter the number to search");

scanf("%d",&x);

for (i=0;i<n;i++)

{

if(ptr+i==x)

flag=1;

}

if (flag==1)

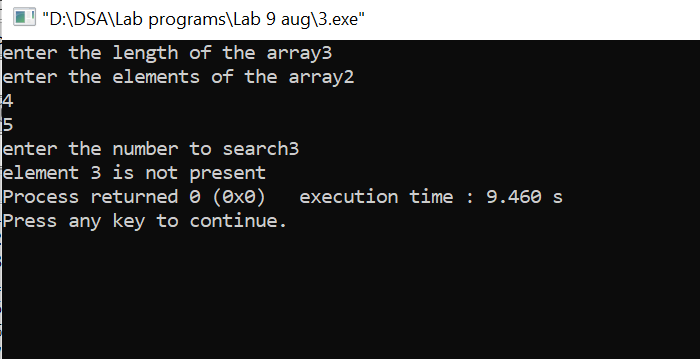
printf("element %d is present",x);

else

printf("element %d is not present",x);

return 0;

}



4.

#include<stdio.h>

#include<conio.h>

struct student

{

char name[50];

int class;

float marks;

int roll;

};

int main()

{

int n;

printf("Enter no. of students : ");

scanf("%d",&n);

struct student s[n],\*ptr;

ptr = s;

int i;

for(i=0;i<n;i++){

printf("enter details of %d student \n",i+1);

fflush(stdin);

printf("enter name : ");

scanf("%s",s[i].name);

printf("Enter class : ");

scanf("%d",&s[i].class);

printf("enter marks obtained : ");

scanf("%f",&s[i].marks);

printf("Enter roll no. : ");

scanf("%d",&s[i].roll);

printf("\n");

}

printf("The enter details are \n");

for(i=0;i<n;i++){

printf("name : ");

puts(s[i].name);

printf("\nclass : ");

printf("%d",s[i].class);

printf("\nmarks obtained : ");

printf("%f",s[i].marks);

printf("\nroll no. : ");

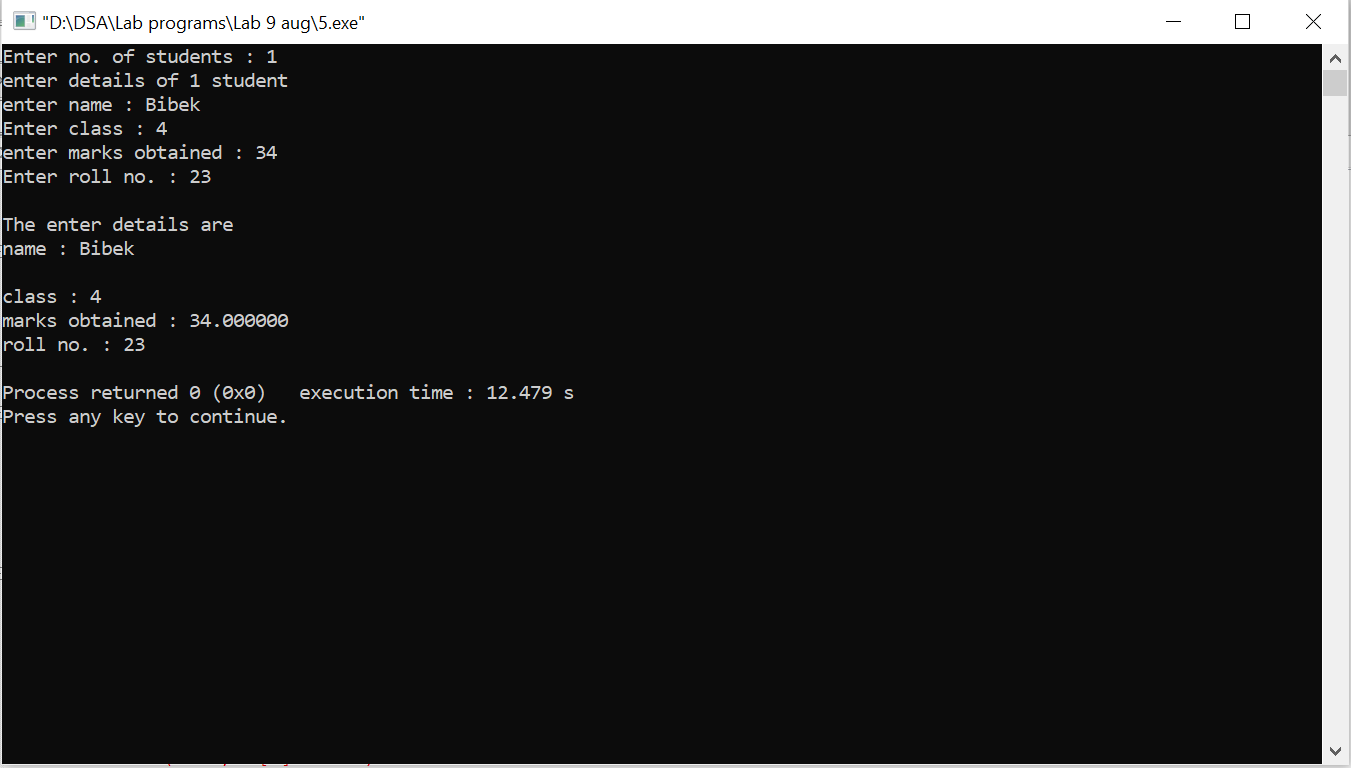
printf("%d",s[i].roll);

printf("\n");

}

return 0;

}



5.

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

int main()

{

int n,i,\*ptr,\*sort;

printf("enter size of array : ");

scanf("%d",&n);

int m=n,j;

ptr = (int\*)malloc(n\*sizeof(int));

sort = (int\*)malloc(n\*sizeof(int));

printf("Enter elements in array : ");

for(i=0;i<n;i++)

scanf("%d",&ptr[i]);

for(i=0,j=0;i<n;i++){

if(\*(ptr+i)%2==0){

sort[j]=\*(ptr+i);

j++;

}

else{

sort[m-1]=\*(ptr+i);

m--;

}

}

printf("The Array after arranging : \n");

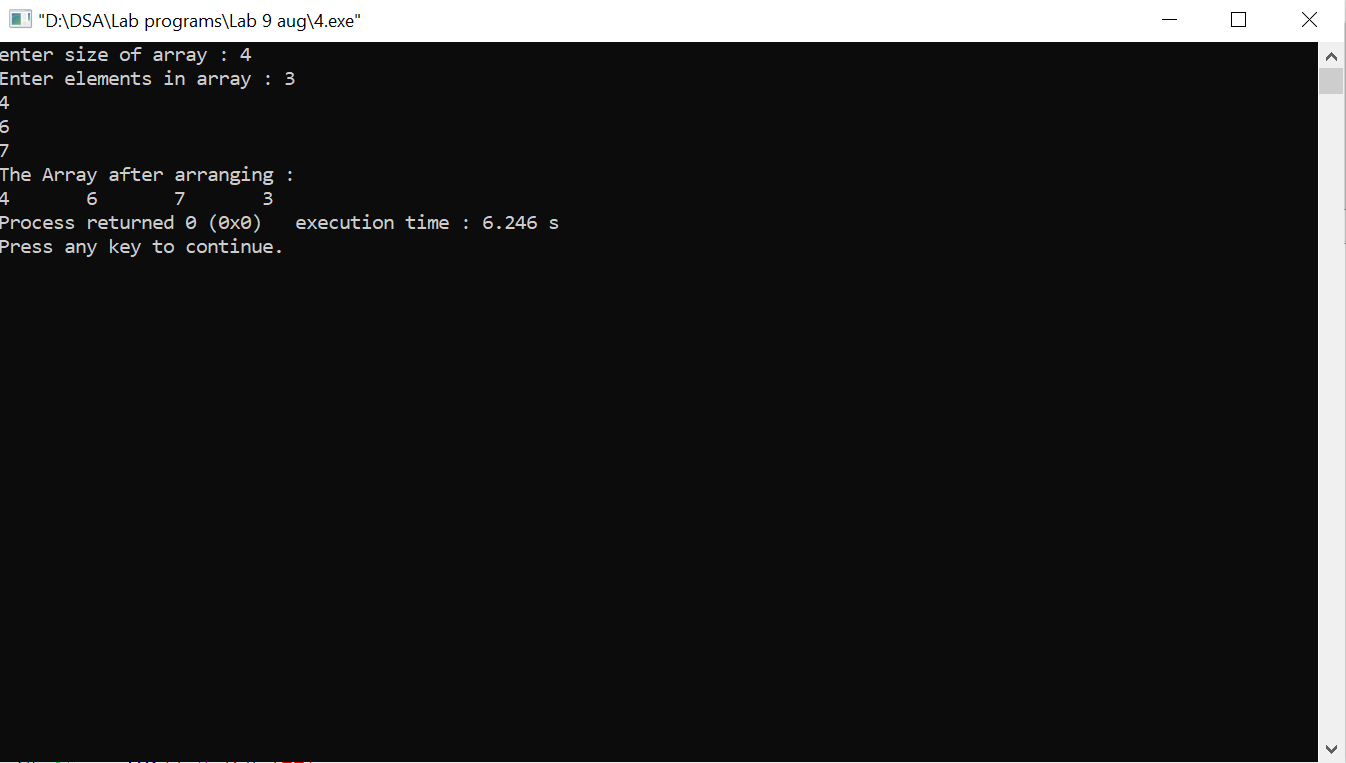
for(i=0;i<n;i++)

printf("%d\t",sort[i]);

return 0;

getch();

}



LAB-4

DATE-08-16

1.

#include <iostream>

using namespace std;

int main()

{

int a[12];

for (int i = 0; i < 12; i++)

{

cout << "\nEnter the "<< i + 1 <<" Elements in an array :";

cin >> a[i];

}

cout <<"\nArray :";

for (int i = 0; i < 12; i++)

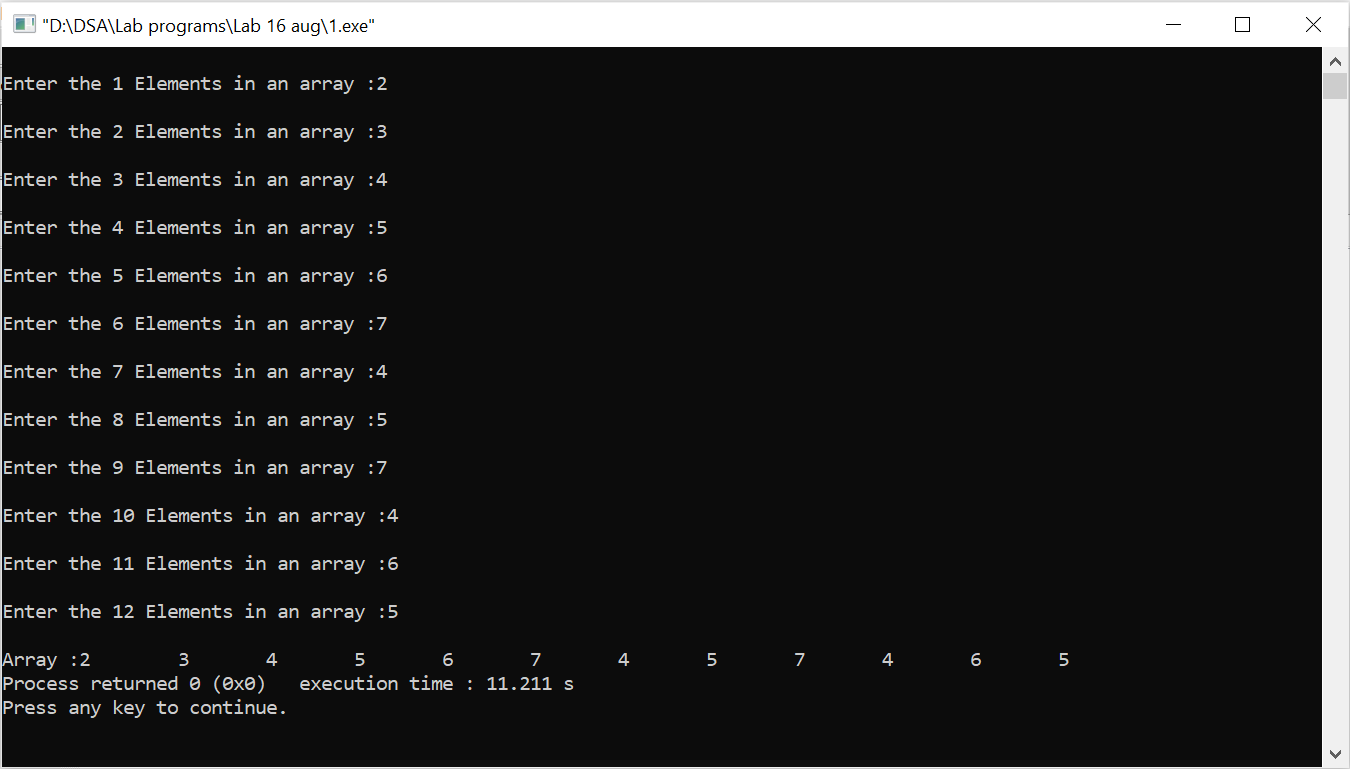
{

cout << a[i]<<"\t";

}

return 0;

}



2.

#include<iostream>

using namespace std;

int main()

{

int k,a,b[100],n;

cout<<"enter the length of an array\t";

cin>>n;

cout<<"enter the elements of an array\n"<<endl;

for(int i=0;i<n;i++)

cin>>b[i];

cout<<"Enter the value of k\t"<<endl;

cin>>k;

cout<<"Enter the value of Kth element\t";

cin>>a;

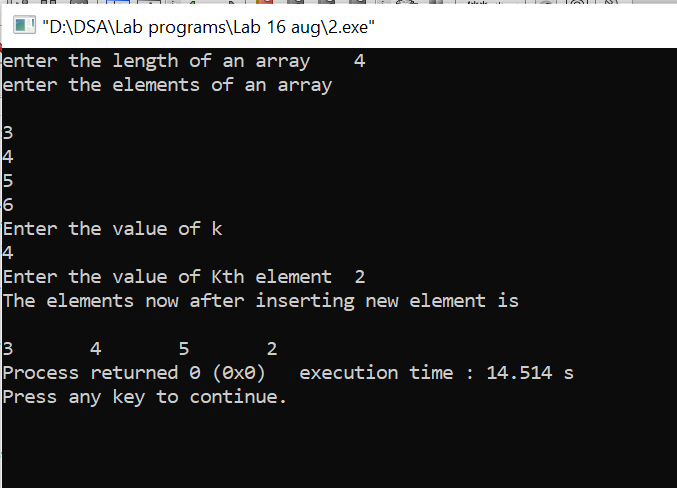
b[k-1]=a;

cout<<"The elements now after inserting new element is \n"<<endl;

for(int i=0;i<n;i++)

cout<<b[i]<<"\t";

}



3.

#include<iostream>

using namespace std;

int main()

{

int k=0,j,b[100],n;

cout<<"enter the length of an array\t";

cin>>n;

cout<<"enter the elements of an array"<<endl;

for(int i=0;i<n;i++)

cin>>b[i];

j=n-1;

for(int i=0;i<j;i++){

k=b[i];

b[i]=b[j];

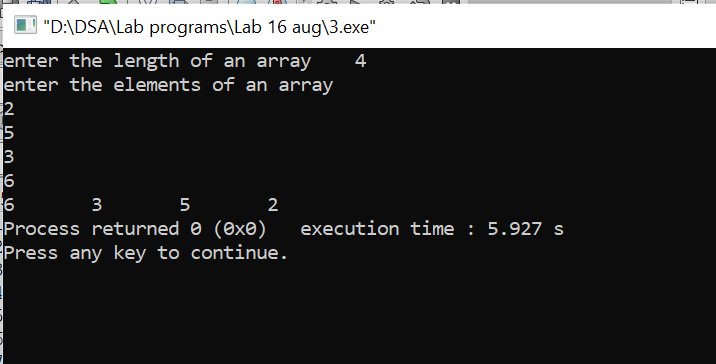
b[j]=k;

j--;

}

for(int i=0;i<n;i++)

cout<<b[i]<<"\t";

}

4.

#include<iostream>

using namespace std;

int main()

{

int k=0,j=0,b[100],n;

cout<<"enter the length of an array\t";

cin>>n;

cout<<"enter the elements of an array"<<endl;

for(int i=0;i<n;i++)

cin>>b[i];

for(int i=0;i<n-1;i++){

for(j=0;j<n-i-1;j++)

{

if (b[j]>b[j+1]){//for decreasing order use '<'

k=b[j];

b[j]=b[j+1];

b[j+1]=k;

}

}

}

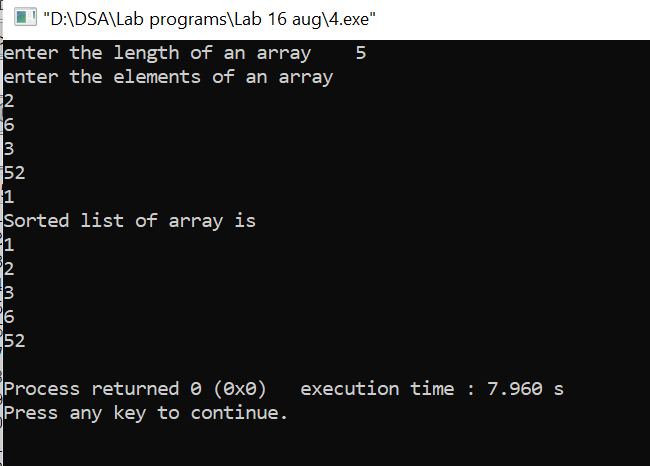
cout<<"Sorted list of array is\n";

for(int i=0;i<n;i++)

cout<<b[i]<<"\n";

return 0;

}



5.

#include<iostream>

using namespace std;

int main()

{

int k=0,l,j,a[100],b[100],n,m,c[100];

cout<<"enter the length of 1st array\t";

cin>>n;

cout<<"enter the elements of 1st array"<<endl;

for(int i=0;i<n;i++)

cin>>a[i];

cout<<"enter the length of 2nd array\t";

cin>>m;

cout<<"enter the elements of 2nd array"<<endl;

for(int i=0;i<m;i++)

cin>>b[i];

j=n+m;

l=0;

for (k=0;k<j;k++)

{

if(k<n)

c[k]=a[k];

else

{

c[k]=b[l];

l++;

}

}

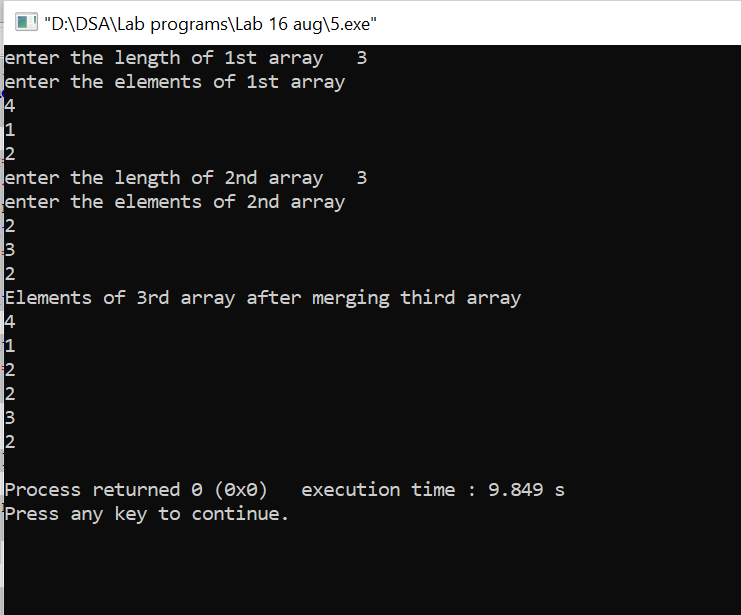
cout<<"Elements of 3rd array after merging third array\n";

for (k=0;k<j;k++)

cout<<c[k]<<"\n";

return 0;

}



6.

#include<iostream>

using namespace std;

int main()

{

int k,a=0,b[100],n;

cout<<"enter the length of an array\t";

cin>>n;

cout<<"enter the elements of an array"<<endl;

for(int i=0;i<n;i++)

cin>>b[i];

cout<<"Enter the element to be searched\t"<<endl;

cin>>k;

for(int i=0;i<n;i++)

{

if(k==b[i]){

cout<<"The element is found at "<<i+1<<" position\n";

a=1;

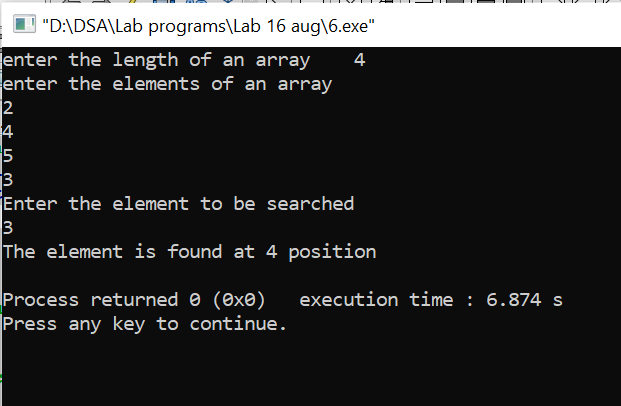
}

}

if (a==0)

cout<<"The element is not found in this array";

}



LAB-5

DATE-08-23

1.

#include<iostream>

#include<conio.h>

using namespace std;

void display(int n){

int p1[100];

cout<<"Enter the coffecient of the polynomial"<<endl;

for (int i=0;i<=n;i++)

cin>>p1[i];

cout<<"The polynomial is ";

for (int i=n;i>=0; i--)

{

if(p1[i]!=0){

cout << p1[i];

if (i!=0)

cout << "x^" << i ;

if (i!=0)

cout << " + ";

}

}

cout<<endl;

}

int main(){

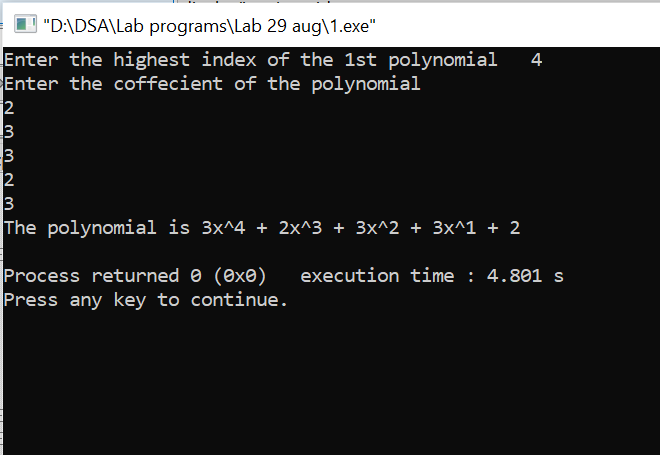
int y;

cout<<"Enter the highest index of the 1st polynomial\t";

cin>>y;

display(y);

}



2.

#include<stdio.h>

void getvalues (int ar[], int n)

{

//truct polynomial P;

for (int i = n; i >= 0; i--)

{

printf ("Enter value of cofficient x^%d:", i);

scanf ("%d", &ar[i]);

}

}

void print (int ar[], int n)

{

for (int i = 0; i <= n; i++)

{

printf ("%d", ar[i]);

if (i != 0)

printf ("x^%d", i);

if (i != n)

printf (" + ");

}

}

void calculate (int ar[], int br[], int n, int m)

{

int cr[200];

if (m >= n)

{

for (int i = 0; i <= n; i++)

cr[i] = ar[i];

for (int i = 0; i <= m; i++)

cr[i] += br[i];

print (cr, m);

}

else if (n > m)

{

for (int i = 0; i <= n; i++)

cr[i] = ar[i];

for (int i = 0; i <= m; i++)

cr[i] += br[i];

print (cr, n);

}

//return cr[];

}

int main ()

{

//struct polynomial P1, P2, P3,P4;

int n, m;

int ar[100], br[100], cr[200];

printf ("Enter highest degree n for Polynomial %d: ", 1);

scanf ("%d", &n);

getvalues (ar, n);

printf ("Enter highest degree n for Polynomial %d: ", 2);

scanf ("%d", &m);

getvalues (br, m);

print (ar, n);

printf ("\n");

print (br, m);

printf ("\n");

printf ("Addition of two polynomial is \n");

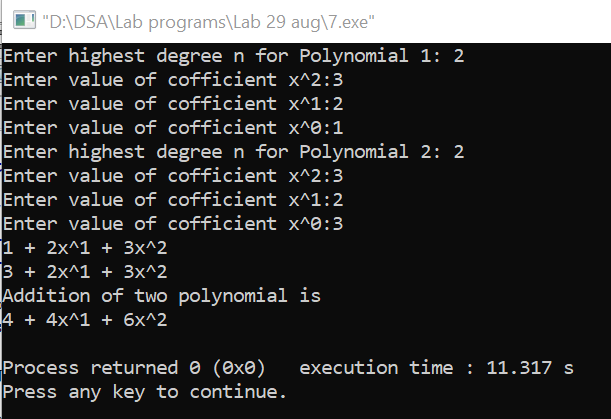
calculate (ar, br, n, m);

printf ("\n");

//system ("PAUSE");

return 0;

}



3.

#include<iostream>

#include<conio.h>

using namespace std;

void display(int n,int m,int p3[]){

int p1[100],p2[100];

cout<<"Enter the coffecient of the polynomial"<<endl;

for (int i=0;i<=n;i++)

cin>>p1[i];

cout<<"The 1st polynomial is ";

for (int i=0;i<=n; i++)

{

cout << p1[i];

if (i != 0)

cout << "x^" << i ;

if (i != n)

cout << " + ";

}

cout<<endl;

cout<<"Enter the coffecient of the polynomial"<<endl;

for (int i=0;i<=m;i++)

cin>>p2[i];

cout<<"THe 2nd polynomial is ";

for (int i=0; i<=m; i++)

{

cout << p2[i];

if (i != 0)

cout << "x^" << i ;

if (i != m)

cout << " + ";

}

cout<<endl;

for (int i = 0; i<=m+n; i++)

p3[i] = 0;

for (int i=0;i<=m;i++)

for (int j=0;j<=n;j++)

p3[i+j]=p1[j]\*p2[i]+p3[i+j];

}

int main(){

int x,y,a[100];

cout<<"Enter the highest index of the 1st polynomial\t";

cin>>y;

cout<<"\nEnter the highest index of the 2nd polynomial\t";

cin>>x;

display(y,x,a);

cout<<"\nAfter multiplication of 1st and 2nd polynomial "<<endl;

for (int i=0; i<=x+y; i++)

{

cout << a[i];

if (i != 0)

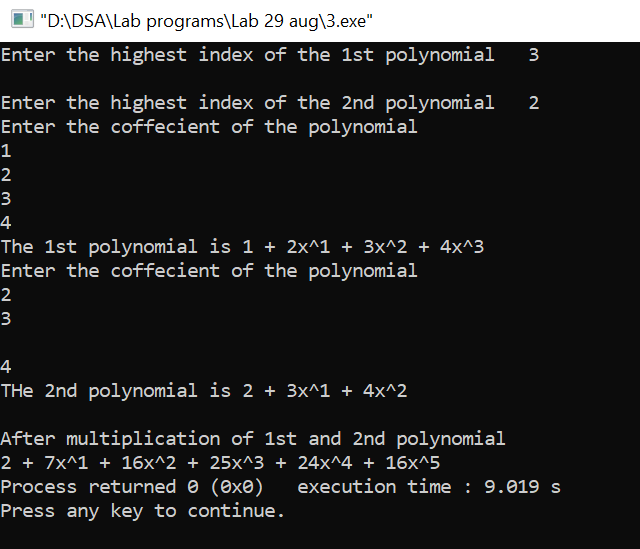
cout << "x^" << i ;

if (i != x+y)

cout << " + ";

}

}



4.

#include<stdio.h>

#define MAX 1000

void printsparse(int[][3]);

void readsparse(int[][3]);

int main()

{

int b1[MAX][3],b2[MAX][3],m,n;

printf("Enter the size of matrix (rows,columns):");

scanf("%d%d",&m,&n);

b1[0][0]=m;

b1[0][1]=n;

readsparse(b1);

printsparse(b1);

}

void readsparse(int b[MAX][3])

{

int i,t;

printf("\nEnter no. of non-zero elements:");

scanf("%d",&t);

b[0][2]=t;

for(i=1;i<=t;i++)

{

printf("\nEnter the next triple(row,column,value):");

scanf("%d%d%d",&b[i][0],&b[i][1],&b[i][2]);

}

}

void printsparse(int b[MAX][3])

{

int i,n;

n=b[0][2];

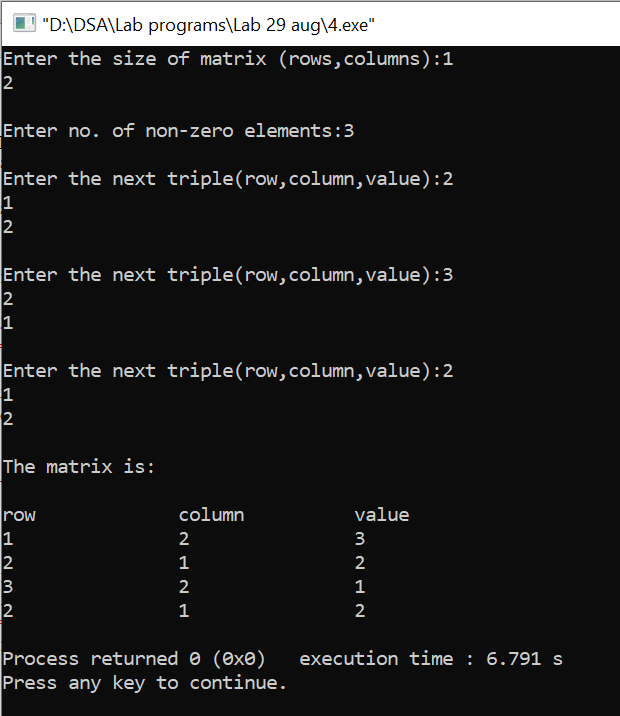
printf("\nThe matrix is:\n");

printf("\nrow\t\tcolumn\t\tvalue\n");

for(i=0;i<=n;i++)

printf("%d\t\t%d\t\t%d\n",b[i][0],b[i][1],b[i][2]);

}



5.

#include<stdio.h>

#define MAX 1000

void printsparse(int[][3]);

void readsparse(int[][3]);

void transpose(int[][3],int[][3]);

int main()

{

int b1[MAX][3],b2[MAX][3],m,n;

printf("Enter the size of matrix (rows,columns):");

scanf("%d%d",&m,&n);

b1[0][0]=m;

b1[0][1]=n;

readsparse(b1);

transpose(b1,b2);

printsparse(b2);

}

void readsparse(int b[MAX][3])

{

int i,t;

printf("\nEnter no. of non-zero elements:");

scanf("%d",&t);

b[0][2]=t;

for(i=1;i<=t;i++)

{

printf("\nEnter the next triple(row,column,value):");

scanf("%d%d%d",&b[i][0],&b[i][1],&b[i][2]);

}

}

void printsparse(int b[MAX][3])

{

int i,n;

n=b[0][2];

printf("\nAfter Transpose:\n");

printf("\nrow\t\tcolumn\t\tvalue\n");

for(i=0;i<=n;i++)

printf("%d\t\t%d\t\t%d\n",b[i][0],b[i][1],b[i][2]);

}

void transpose(int b1[][3],int b2[][3])

{

int i,j,k,n;

b2[0][0]=b1[0][1];

b2[0][1]=b1[0][0];

b2[0][2]=b1[0][2];

k=1;

n=b1[0][2];

for(i=0;i<b1[0][1];i++)

for(j=1;j<=n;j++)

if(i==b1[j][1])

{

b2[k][0]=i;

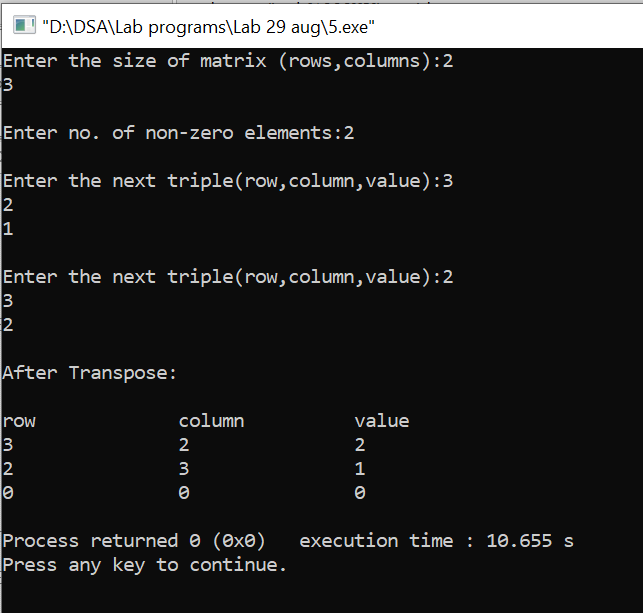
b2[k][1]=b1[j][0];

b2[k][2]=b1[j][2];

k++;

}

}



6.

#include<stdio.h>

#include<stdlib.h>

#define MAX 20

void printsparse(int b[MAX][3]);

void readsparse(int b[MAX][3]);

void addsparse(int b1[MAX][3],int b2[MAX][3],int b3[MAX][3]);

void main()

{

int b1[MAX][3],b2[MAX][3],b3[MAX][3];

readsparse(b1);

readsparse(b2);

addsparse(b1,b2,b3);

printsparse(b3);

}

void readsparse(int b[MAX][3])

{

int i,t,m,n;

printf("\nEnter no. of rows and columns:");

scanf("%d%d",&m,&n);printf("No. of non-zero triples:");

scanf("%d",&t);

b[0][0]=m;

b[0][1]=n;

b[0][2]=t;

for(i=1;i<=t;i++)

{

printf("Enter the triples(row,column,value):");

scanf("%d%d%d",&b[i][0],&b[i][1],&b[i][2]);

}

}

void addsparse(int b1[MAX][3],int

b2[MAX][3],int b3[MAX][3])

{

int t1,t2,i,j,k;

if(b1[0][0]!=b2[0][0]||b1[0][1]!=b2[0][1])

{

printf("nYou have entered invalid matrix!!Size must be equal");

exit(0);

}

t1=b1[0][2];

t2=b2[0][2];

i=j=k=0;

b3[0][0]=b1[0][0];

b3[0][1]=b1[0][1];

while(i<=t1&&j<=t2)

{

if(b1[i][0]<b2[j][0])

{

b3[k][0]=b1[i][0];

b3[k][1]=b1[i][1];

b3[k][2]=b1[i][2];

k++;

i++;

}

else if(b2[j][0]<b1[i][0])

{

b3[k][0]=b2[j][0];

b3[k][1]=b2[j][1];

b3[k][2]=b2[j][2];

k++;

j++;

}

else if(b1[i][1]<b2[j][1])

{

b3[k][0]=b1[i][0];

b3[k][1]=b1[i][1];

b3[k][2]=b1[i][2];

k++;

i++;

}

else if(b2[j][1]<b1[i][1])

{

b3[k][0]=b2[j][0];

b3[k][1]=b2[j][1];

b3[k][2]=b2[j][2];

k++;

j++;

}

else

{

b3[k][0]=b1[i][0];

b3[k][1]=b1[i][1];

b3[k][2]=b1[i][2]+b2[j][2];

k++;

i++;

j++;

}

}

while(i<=t1)

{

b3[k][0]=b1[i][0];

b3[k][1]=b1[i][1];

b3[k][2]=b1[i][2];

i++;

k++;

}

while(j<=t2)

{

b3[k][0]=b2[j][0];

b3[k][1]=b1[j][1];

b3[k][2]=b1[j][2];

j++;

k++;

}

b3[0][2]=k-1;

}

void printsparse(int b[MAX][3])

{

int i,t;

t=b[0][2];

printf("\nrow\tcolumn\tvalue");

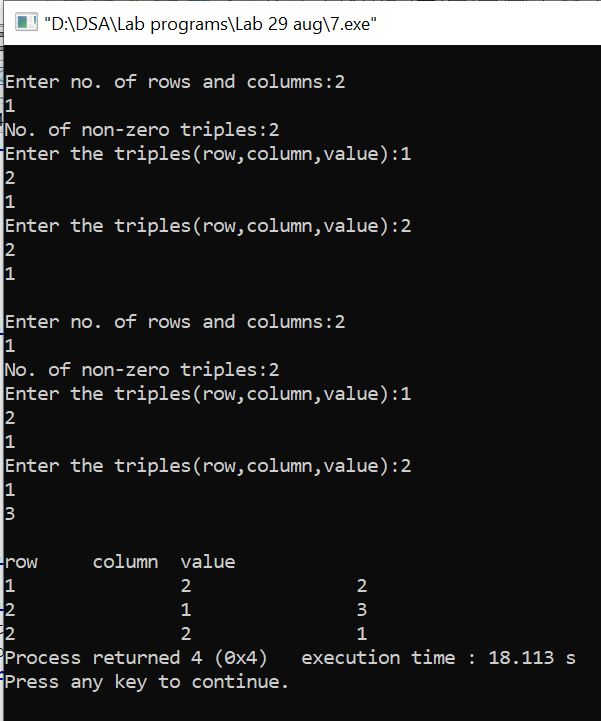
for(i=1;i<=t;i++)

{

printf("\n%d\t\t%d\t\t%d",b[i][0],b[i][1],b[i][2]);

}

}



LAB-6

DATE-09-05

1.

#include <stdio.h>

#include <stdlib.h>

struct node

{

int info;

struct node \*link;

};

struct node \*start = NULL;

void createList ()

{

int n;

printf ("\nEnter the total number of nodes: ");

scanf ("%d", &n);

struct node \*newNode, \*t;

int data, info, i;

start = (struct node \*) malloc (sizeof (struct node));

if (start == NULL)

{

printf ("Unable to allocate memory.");

exit (0);

}

printf ("Enter the data of node 1: ");

scanf ("%d", &start->info);

start->link = NULL;

t = start;

for (i = 2; i <= n; i++)

{

newNode = (struct node \*) malloc (sizeof (struct node));

if (newNode == NULL)

{

printf ("Unable to allocate memory.");

break;

}

printf ("Enter the data of node %d: ", i);

scanf ("%d", &info);

newNode->info = info;

newNode->link = NULL;

t->link = newNode;

t = t->link;

}

}

void traverse ()

{

if (start == NULL)

{

printf ("\nList is empty\n");

}

struct node \*temp=start;

while (temp->link != NULL)

{

printf("%d ", temp->info);

temp = temp->link;

}

printf("%d",temp->info);

}

void

insertAtFront ()

{

int data;

struct node \*temp;

temp = malloc (sizeof (struct node));

printf ("\nEnter number to" " be inserted : ");

scanf ("%d", &data);

temp->info = data;

temp->link = start;

start = temp;

}

void insertAtEnd ()

{

int data;

struct node \*temp, \*head;

temp = malloc (sizeof (struct node));

printf ("\nEnter number to" " be inserted : ");

scanf ("%d", &data);

// Changes links

temp->link = 0;

temp->info = data;

head = start;

while (head->link != NULL)

{

head = head->link;

}

head->link = temp;

}

void

insertAtPosition ()

{

struct node \*temp, \*newnode;

int pos, data, i = 1;

newnode = malloc (sizeof (struct node));

// Enter the position and data

printf ("\nEnter position and data :");

scanf ("%d %d", &pos, &data);

temp = start;

newnode->info = data;

newnode->link = 0;

while (i < pos - 1)

{

temp = temp->link;

i++;

}

newnode->link = temp->link;

temp->link = newnode;

}

void

deleteFirst ()

{

struct node \*temp;

if (start == NULL)

printf ("\nList is empty\n");

else

{

temp = start;

start = start->link;

free (temp);

}

}

void

deleteEnd ()

{

struct node \*temp, \*prevnode;

if (start == NULL)

printf ("\nList is Empty\n");

else

{

temp = start;

while (temp->link != 0)

{

prevnode = temp;

temp = temp->link;

}

free (temp);

prevnode->link = 0;

}

}

int main ()

{

int choice;

while (1)

{

printf ("\n\t1 To CREATE LIST\n");

printf ("\t2 To TRAVERSE\n");

printf ("\t3 For INSERTION at starting\n");

printf ("\t4 For INSERTION at end\n");

printf ("\t5 For INSERTION at Kth position\n");

printf ("\t6 For DELETION of first element\n");

printf ("\t7 For DELETION of last element\n");

printf ("\t8 To exit\n");

printf ("\nEnter Choice : ");

scanf ("%d", &choice);

printf("\n");

switch (choice)

{

case 1:

createList ();

break;

case 2:

traverse ();

break;

case 3:

insertAtFront ();

break;

case 4:

insertAtEnd ();

break;

case 5:

insertAtPosition ();

break;

case 6:

deleteFirst ();

break;

case 7:

deleteEnd ();

break;

case 8:

exit (1);

break;

default:

printf ("Incorrect Choice\n");

}

}

return 0;

}

