PREETHA A

POLYNOMIAL MANIPULATION

#include<stdio.h>

struct poly

{

int coeff;

int expo;

};

struct poly p1[10],p2[10],p3[10];

int readPoly(struct poly []);

int addPoly(struct poly [],struct poly [],int ,int ,struct poly []);

void displayPoly( struct poly [],int terms);

int main()

{

int t1,t2,t3;

t1=readPoly(p1);

printf(" \n First polynomial : ");

displayPoly(p1,t1);

t2=readPoly(p2);

printf(" \n Second polynomial : ");

displayPoly(p2,t2);

t3=addPoly(p1,p2,t1,t2,p3);

printf(" \n\n Resultant polynomial after addition : ");

displayPoly(p3,t3);

printf("\n");

return 0;

}

int readPoly(struct poly p[10])

{

int t1,i;

printf("\n\n Enter the total number of terms in the polynomial:");

scanf("%d",&t1);

printf("\n Enter the COEFFICIENT and EXPONENT in DESCENDING ORDER\n");

for(i=0;i<t1;i++)

{

printf(" Enter the Coefficient(%d): ",i+1);

scanf("%d",&p[i].coeff);

printf(" Enter the exponent(%d): ",i+1);

scanf("%d",&p[i].expo);

}

return(t1);

}

int addPoly(struct poly p1[10],struct poly p2[10],int t1,int t2,struct poly p3[10])

{

int i,j,k;

i=0;

j=0;

k=0;

while(i<t1 && j<t2)

{

if(p1[i].expo==p2[j].expo)

{

p3[k].coeff=p1[i].coeff + p2[j].coeff;

p3[k].expo=p1[i].expo;

i++;

j++;

k++;

}

else if(p1[i].expo>p2[j].expo)

{

p3[k].coeff=p1[i].coeff;

p3[k].expo=p1[i].expo;

i++;

k++;

}

else

{

p3[k].coeff=p2[j].coeff;

p3[k].expo=p2[j].expo;

j++;

k++;

}

}

while(i<t1)

{

p3[k].coeff=p1[i].coeff;

p3[k].expo=p1[i].expo;

i++;

k++;

}

while(j<t2)

{

p3[k].coeff=p2[j].coeff;

p3[k].expo=p2[j].expo;

j++;

k++;

}

return(k);

}

void displayPoly(struct poly p[10],int term)

{

int k;

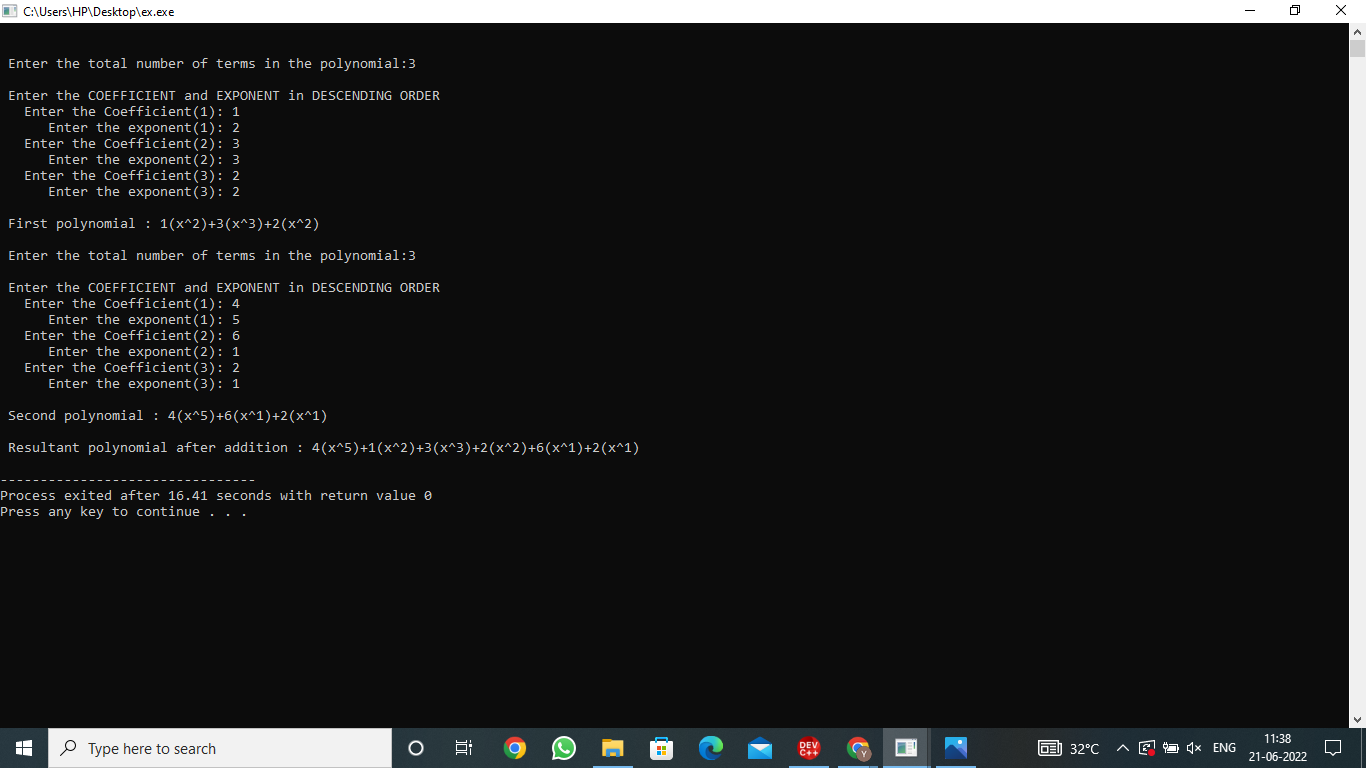
for(k=0;k<term-1;k++)

printf("%d(x^%d)+",p[k].coeff,p[k].expo);

printf("%d(x^%d)",p[term-1].coeff,p[term-1].expo);

}

OUTPUT:



**BUBBLE SORT:**

**#include<stdio.h>**

**int main()**

**{**

**int a[20],i,j,temp,n;**

**printf("\nEnter the number of elements:");**

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

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

**{**

**printf("\nEnter the element [%d]:",i);**

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

**}**

**printf("\nBefore sorting:\n");**

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

**{**

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

**}**

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

**{**

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

**{**

**if(a[j]>a[j+1])**

**{**

**temp=a[j];**

**a[j]=a[j+1];**

**a[j+1]=temp;**

**}**

**}**

**}**

**printf("\nAfter sorting:\n");**

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

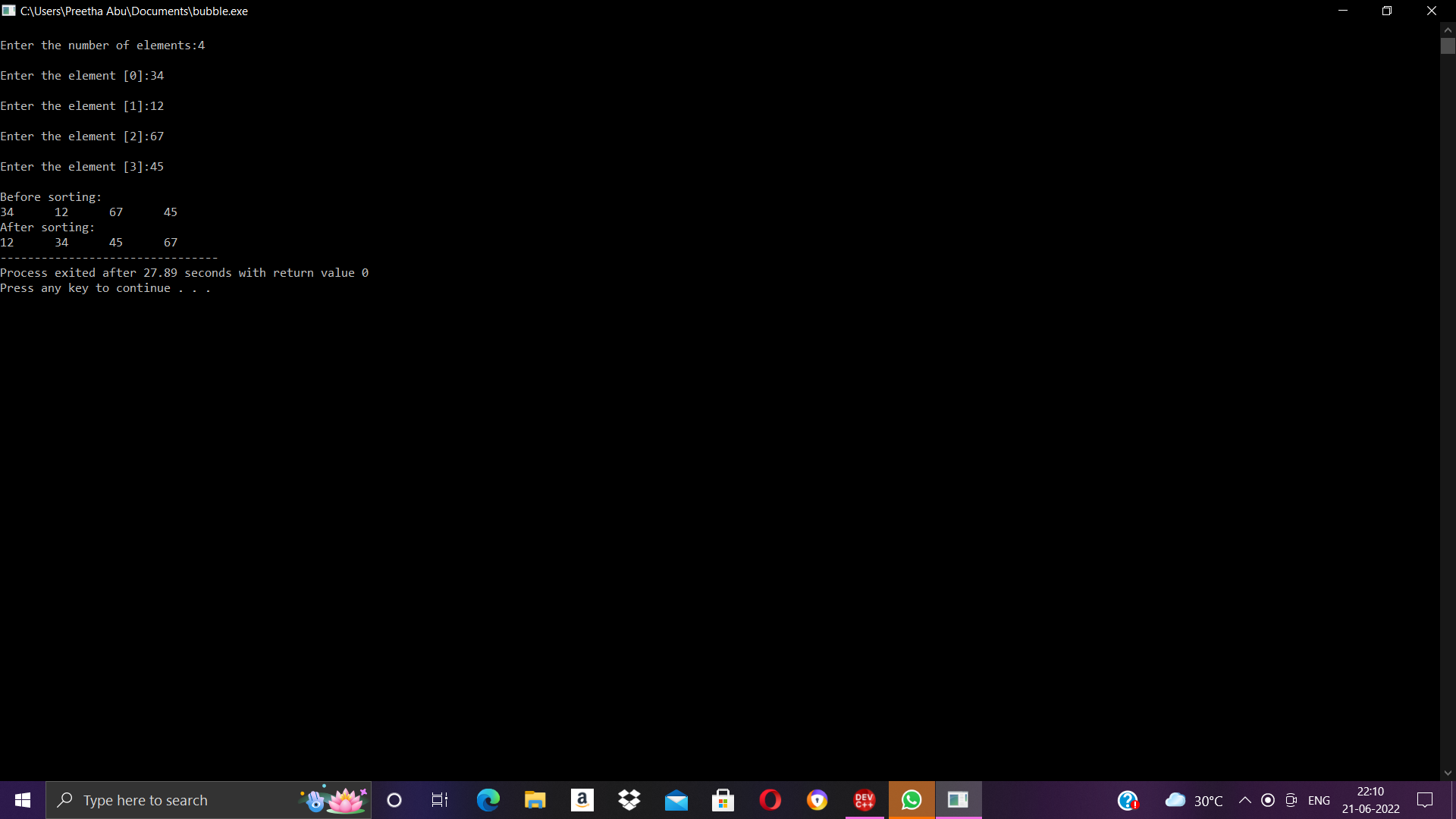
**{**

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

**}**

**return 0;**

**}**



**INSERTION SORT:**

**#include<stdio.h>**

**void main()**

**{**

**int a[20],i,j,n,temp;**

**printf("Enter the number of elements :");**

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

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

**{**

**printf("Enter element %d : ",i);**

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

**}**

**printf(" Unsorted Array \n");**

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

**{**

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

**}**

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

**{**

**temp = a[i];**

**j = i-1;**

**while(j>=0 && a[j]>temp)**

**{**

**a[j+1] = a[j];**

**j--;**

**}**

**a[j+1] = temp;**

**} printf(" \nSorted array");**

**printf("\n");**

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

**{**

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

**}**

**}**

