DATA STRUCTURES LAB

LAB TASK – 1

PROGRAM – 1

#include <stdio.h>

int main()

{

int a[20],sum=0,size,\*ptr;

printf("enter the size of an array :");

scanf("%d",&size);

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

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

{

printf("enter a[%d] element = ",i);

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

}

ptr=a;

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

{

sum = sum + \*ptr;

ptr++;

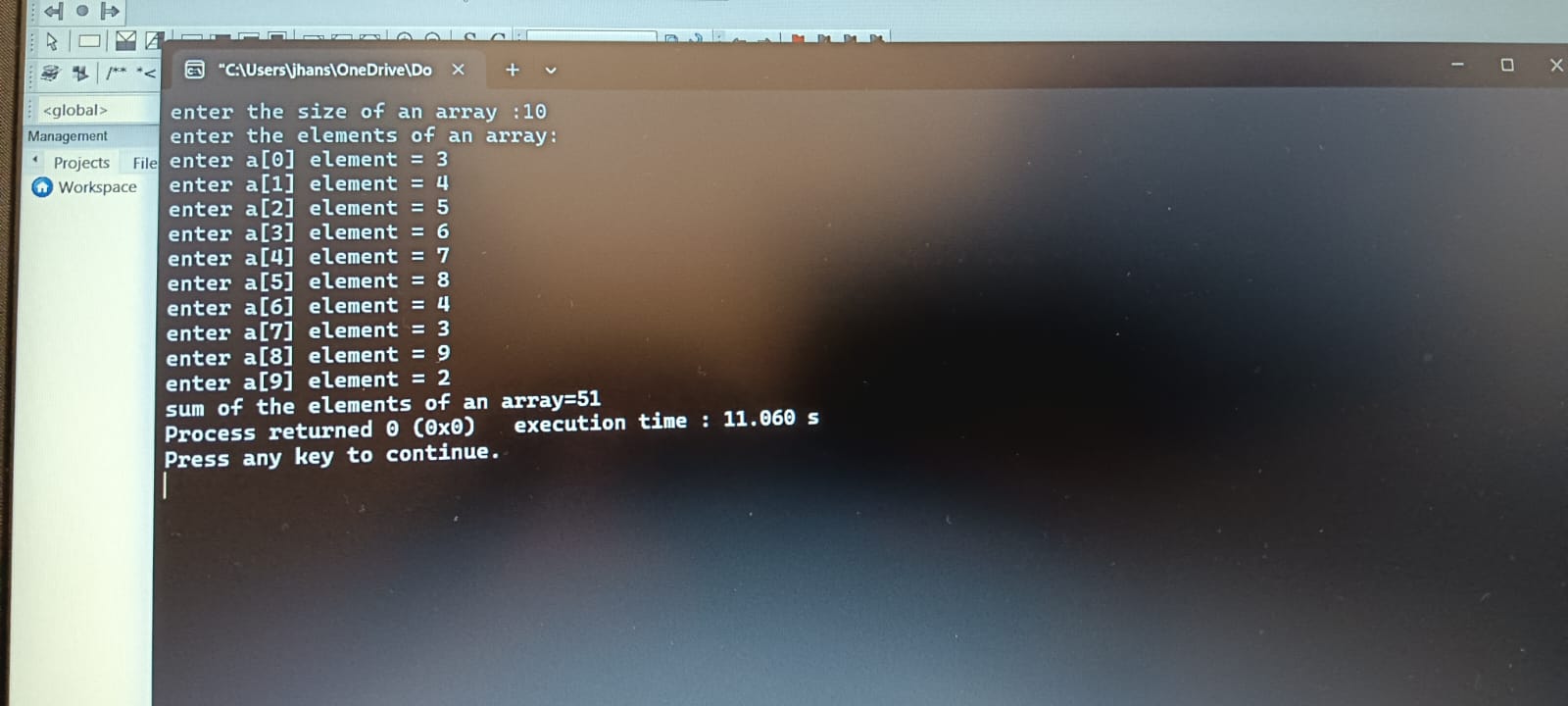
}

printf("sum of the elements of an array=%d",sum);

return 0;

}

OUTPUT :



PROGRAM - 2

#include <stdio.h>

int main()

{

int a,b,\*p,\*q,temp;

printf("before swapping ,\n");

printf("value of a =");

scanf("%d",&a);

printf("value of b =");

scanf("%d",&b);

p=&a;

q=&b;

temp=\*p;

\*p=\*q;

\*q=temp;

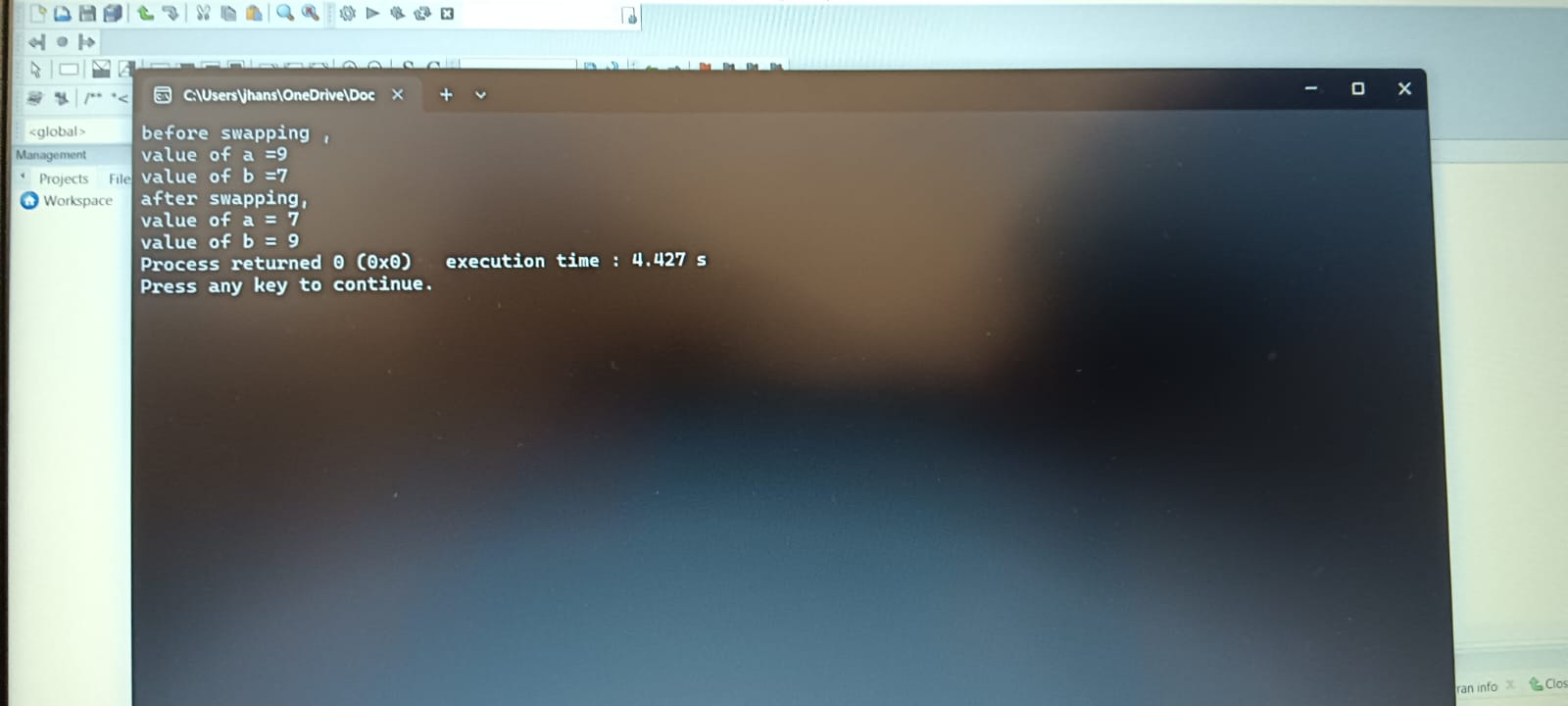
printf("after swapping, \n");

printf("value of a = %d\nvalue of b = %d ",\*p,\*q);

return 0;

}

OUTPUT :



PROGRAM – 3

#include <stdio.h>

#include <string.h>

int main()

{

char str[50],temp;

printf("enter a string: ");

scanf("%s",str);

for(int i=0,j=strlen(str)-1;i<j;i++,j--)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

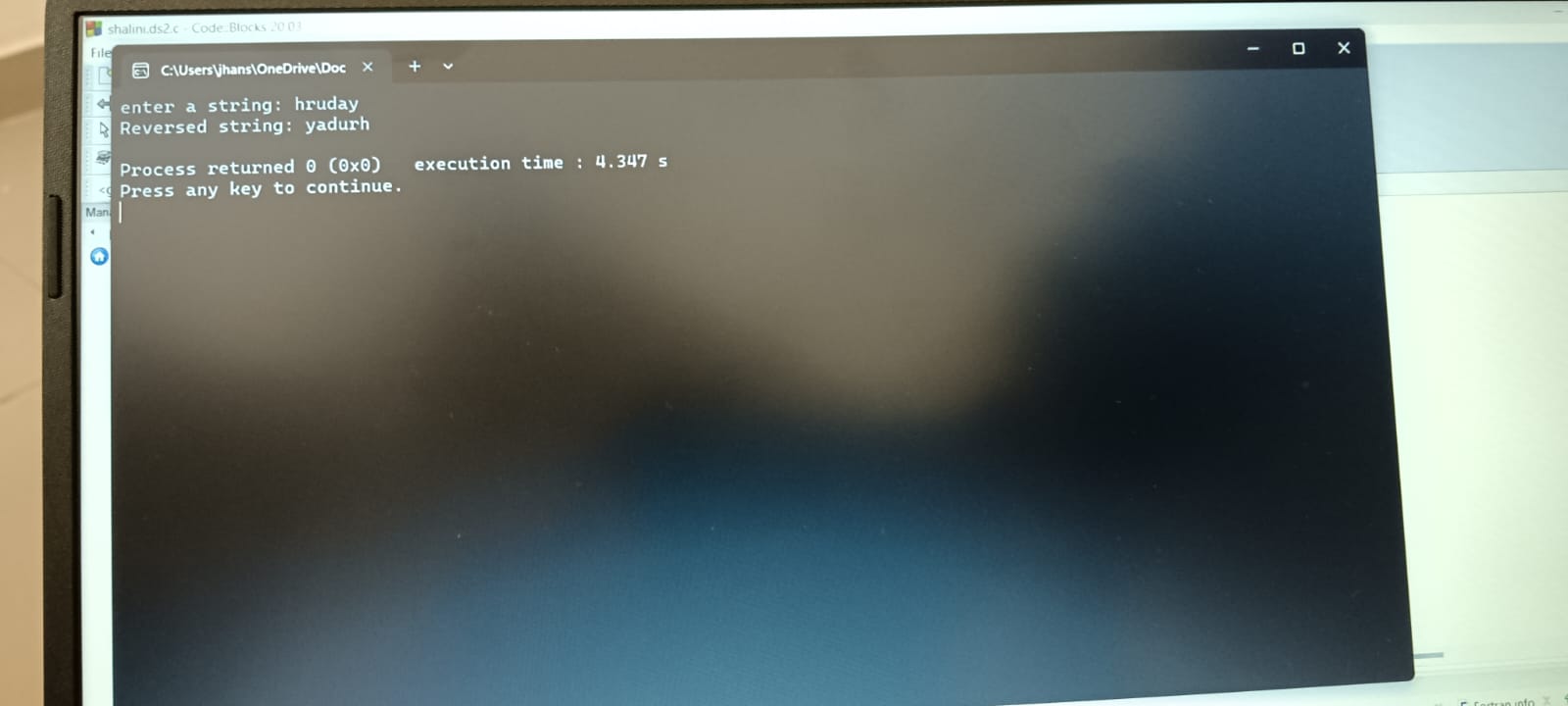
}

printf("Reversed string: %s",str);

return 0;

}

OUTPUT :



PROGRAM – 4

#include <stdio.h>

int power(int base, int exponent);

int calpow(int base, int exponent, int (\*p)(int, int));

int main()

{

int base,result,exponent;

printf("enter base: ");

scanf("%d",&base);

printf("enter exponent: ");

scanf("%d",&exponent);

result = calpow(base,exponent,power);

printf("%d to the power %d = %d\n",base,exponent,result);

return 0;

}

int power(int base, int exponent)

{

int result = 1;

while(exponent > 0)

{

result=result\*base;

exponent--;

}

return result;

}

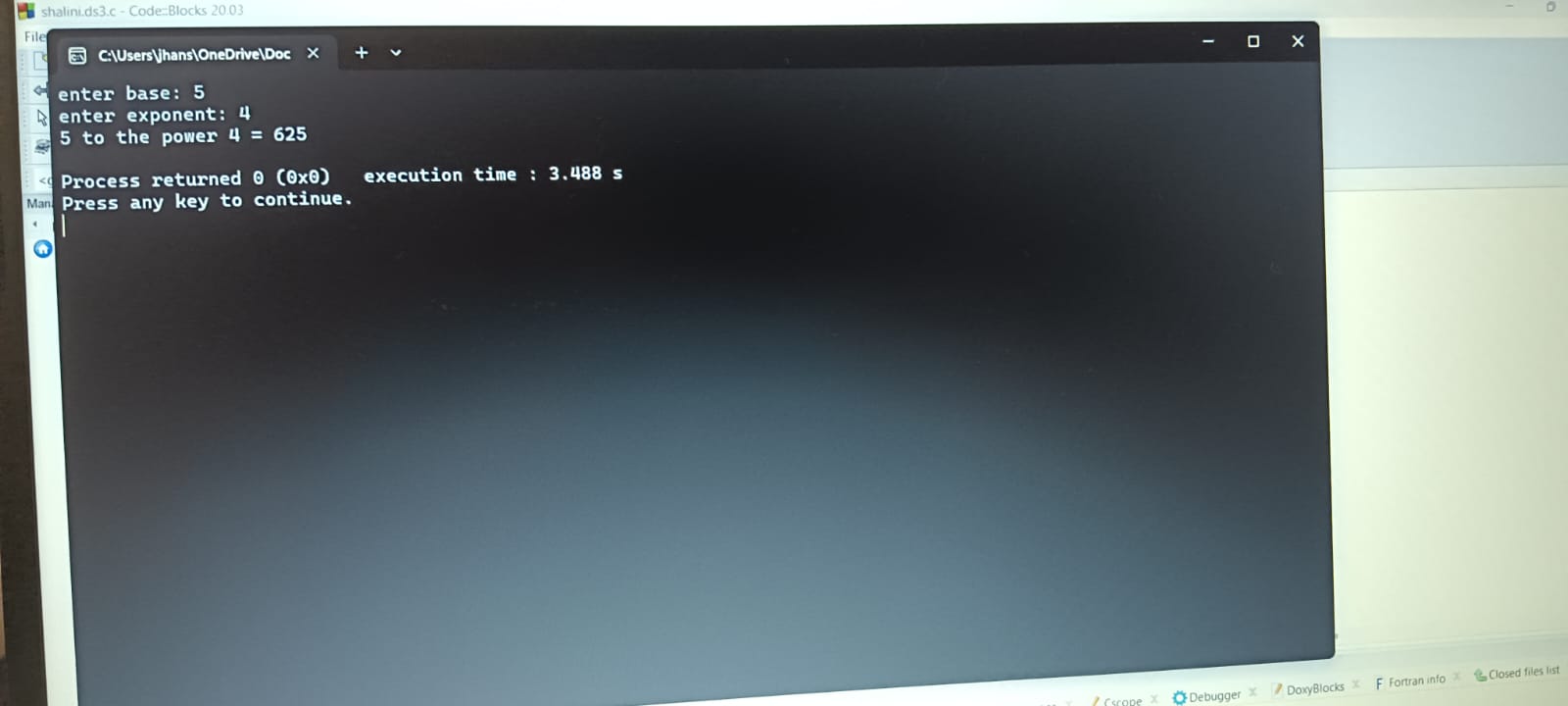
int calpow(int base,int exponent,int (\*p)(int, int))

{

return p(base, exponent);

}

OUTPUT :



PROGRAM – 5

#include <stdio.h>

#include <stdlib.h>

int main()

{

int rows,cols;

printf("enter no. of rows: ");

scanf("%d",&rows);

printf("enter no. of columns: ");

scanf("%d",&cols);

int \*\*matrix=(int\*\*)calloc(rows,sizeof(int \*));

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

{

matrix[i]=(int\*)calloc(cols,sizeof(int));

}

printf("enter elements of the matrix:\n");

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

{

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

{

printf("enter matrix[%d][%d] element = ",i,j);

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

}

}

printf("The entered matrix is as below:\n");

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

{

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

{

printf("%d ",matrix[i][j]);

}

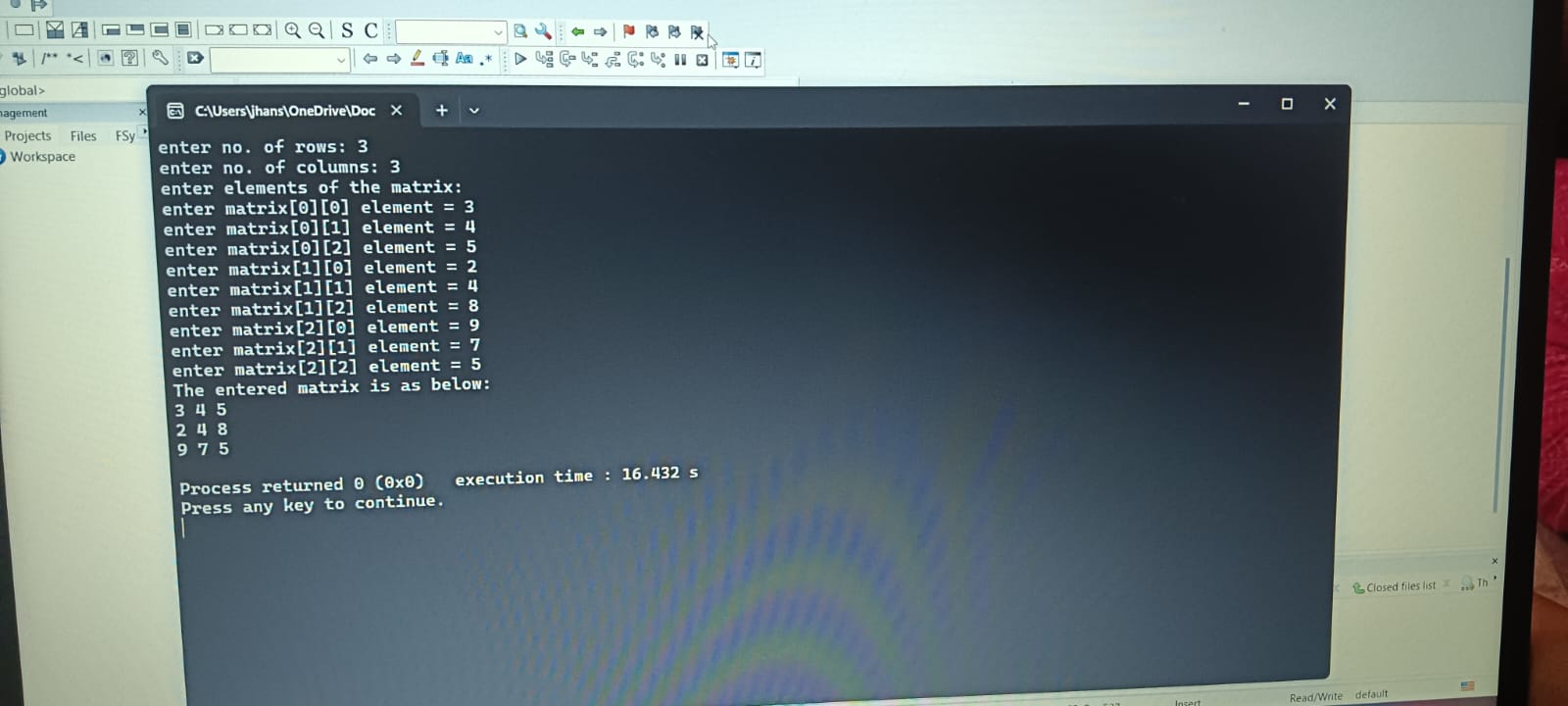
printf("\n");

}

return 0;

}

OUTPUT :



\*\*\*\*\*THE END\*\*\*\*\*