**Set-5**

**Easy**

1.Store,Sort,Print

#include <iostream>

using namespace std;

int main()

{

int a[50],n,i,j,temp;

cout<<"enter size:";

cin>>n;

cout<<"enter elements:";

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

cin>>a[i];

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

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

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

{

temp=a[j];

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

a[j+1]=temp;

}

cout<<"sort:";

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

{

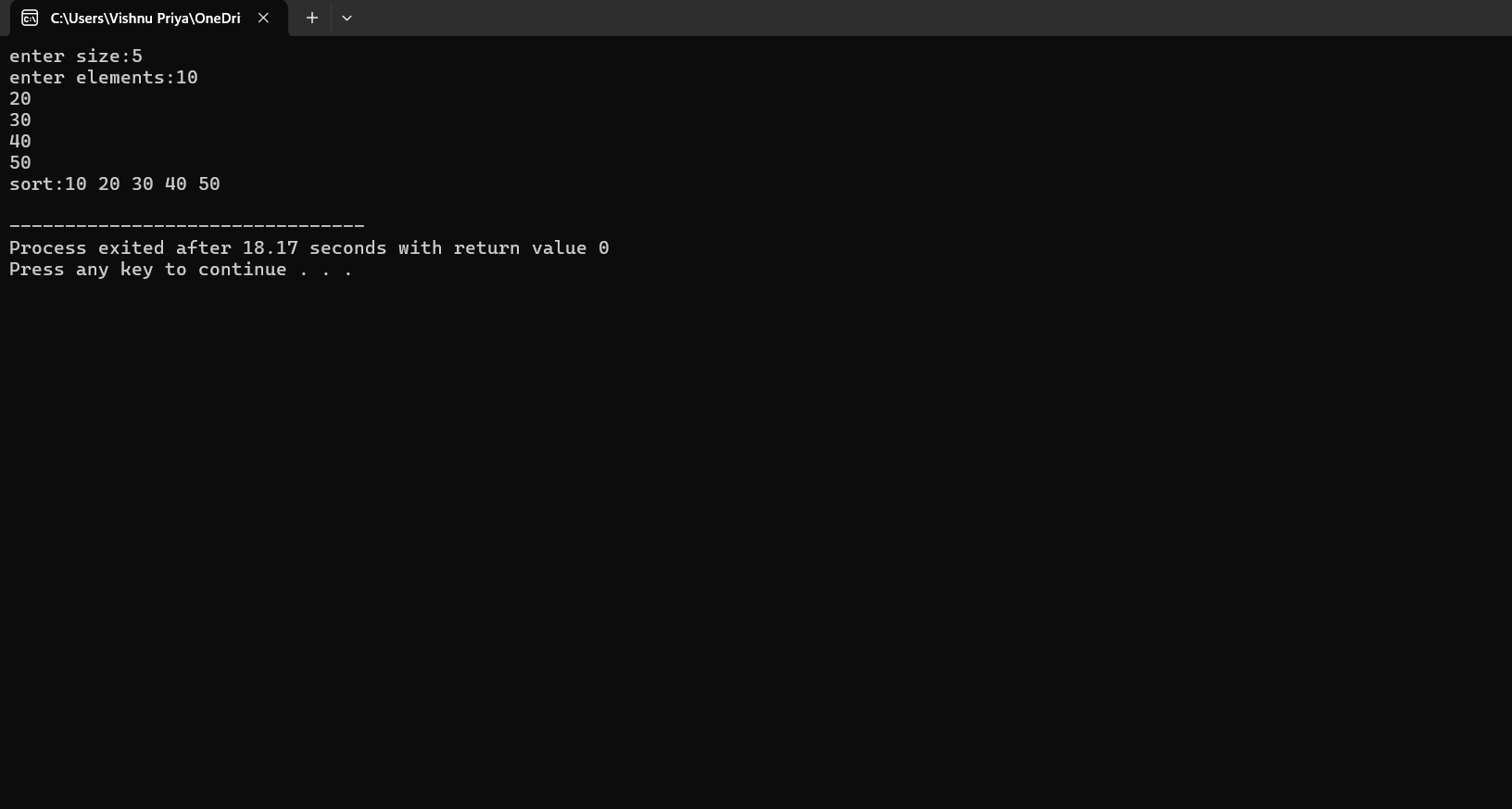
cout<<a[i]<<" ";

}

cout<<endl;

return 0;

}



2.Student class (Ip none op unknown)

#include <iostream>

#include <string>

using namespace std;

class Student

{

public:

string name;

Student(string n = "Unknown")

{

name = n;

}

void printName()

{

cout << name << endl;

}

};

int main() {

Student student1("vishnu");

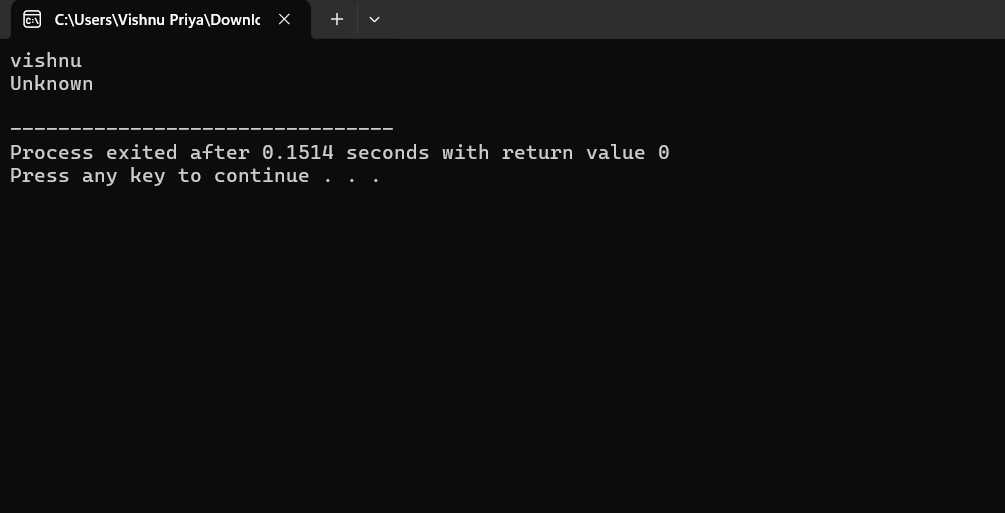
Student student2;

student1.printName();

student2.printName();

return 0;

}



3.Area of Circle,Rectangle,Triangle

#include<iostream>

#include<cmath>

using namespace std;

class Shape {

public:

double areaOfCircle(double radius)

{

return M\_PI \* radius \* radius;

}

double areaOfTriangle(double base, double height)

{

return 0.5 \* base \* height;

}

double areaOfRectangle(double length, double width)

{

return length \* width;

}

};

int main()

{

Shape shape;

double radius;

cout<<"Enter the radius of the circle: ";

cin>>radius;

cout<<"The area of the circle is:"<< shape.areaOfCircle(radius)<<endl;

double base, height;

cout << "Enter the base of the triangle: ";

cin >> base;

cout << "Enter the height of the triangle: ";

cin >> height;

cout << "The area of the triangle is: " << shape.areaOfTriangle(base, height) << endl;

double length, width;

cout << "Enter the length of the rectangle: ";

cin>>length;

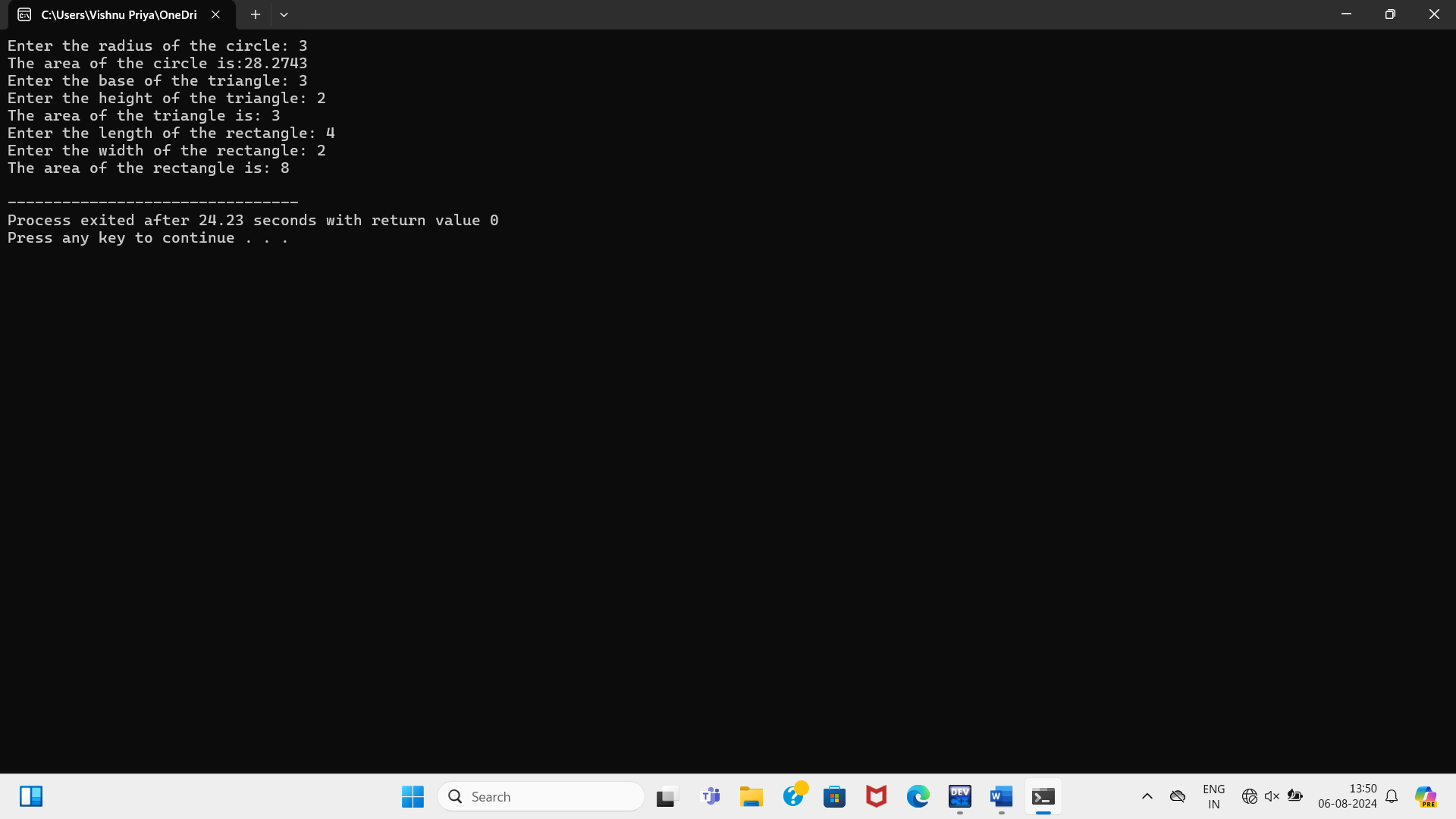
cout<<"Enter the width of the rectangle: ";

cin >> width;

cout << "The area of the rectangle is: " << shape.areaOfRectangle(length, width) << endl;

return 0;

}



4.Reverse a string using pointer

#include <iostream>

using namespace std;

void reverseString(const char\*str)

{

int length = 0;

const char\* p = str;

while (\*p!='\0')

{

length++;

p++;

}

cout<<"Reversed string:";

for (int i = length-1; i>= 0;i--)

{

cout << \*(str+i);

}

cout << endl;

}

int main()

{

const int max\_length = 100;

char input[max\_length];

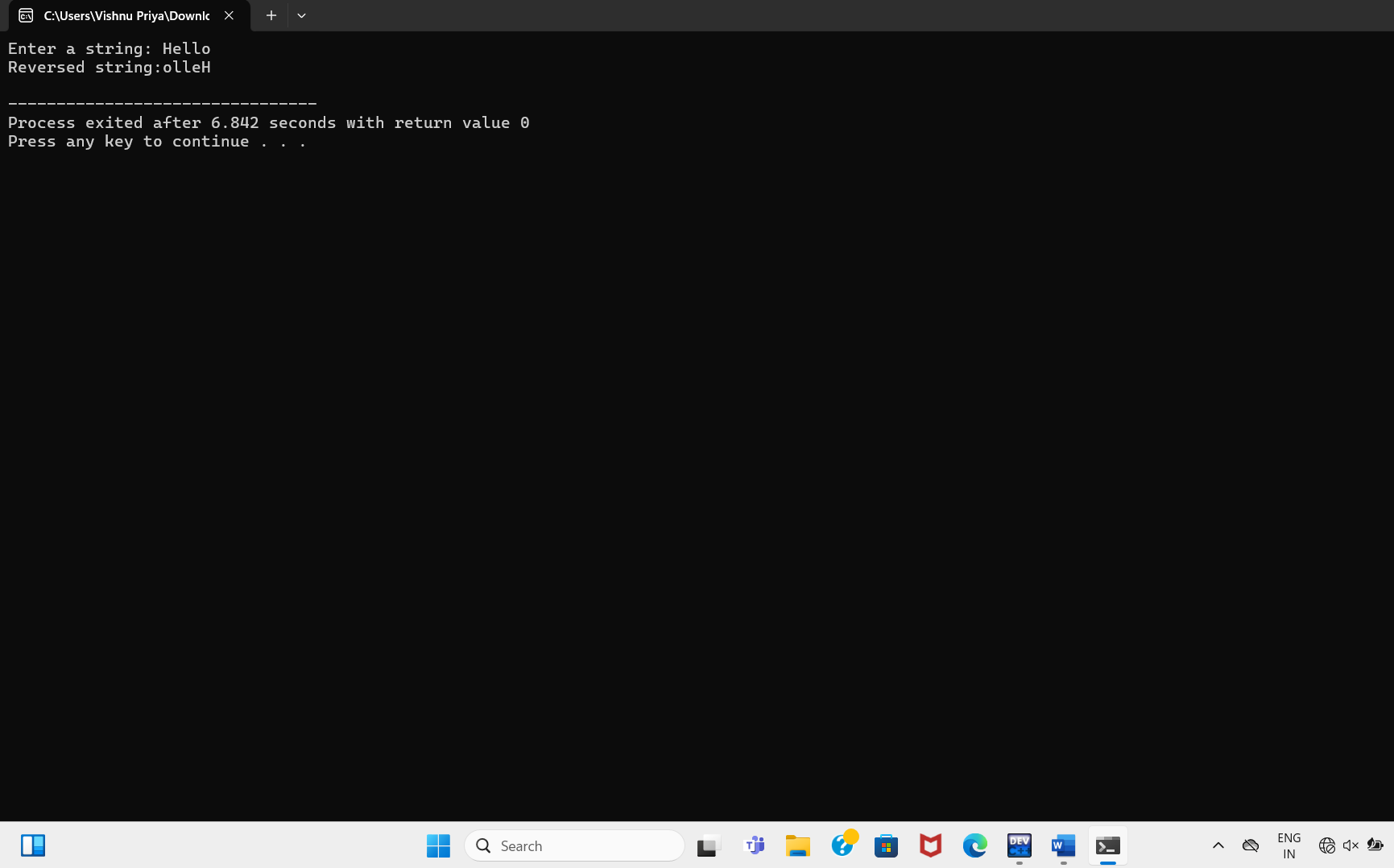
cout << "Enter a string: ";

cin.getline(input, max\_length);

reverseString(input);

return 0;

}



5. Palindrome or not

#include<iostream>

using namespace std;

int main()

{

int n, reversed=0,remainder,original;

cout<<"Enter an integer: ";

cin>>n;

original=n;

while (n!=0)

{

remainder=n % 10;

reversed=reversed\*10+remainder;

n/=10;

}

if (original==reversed)

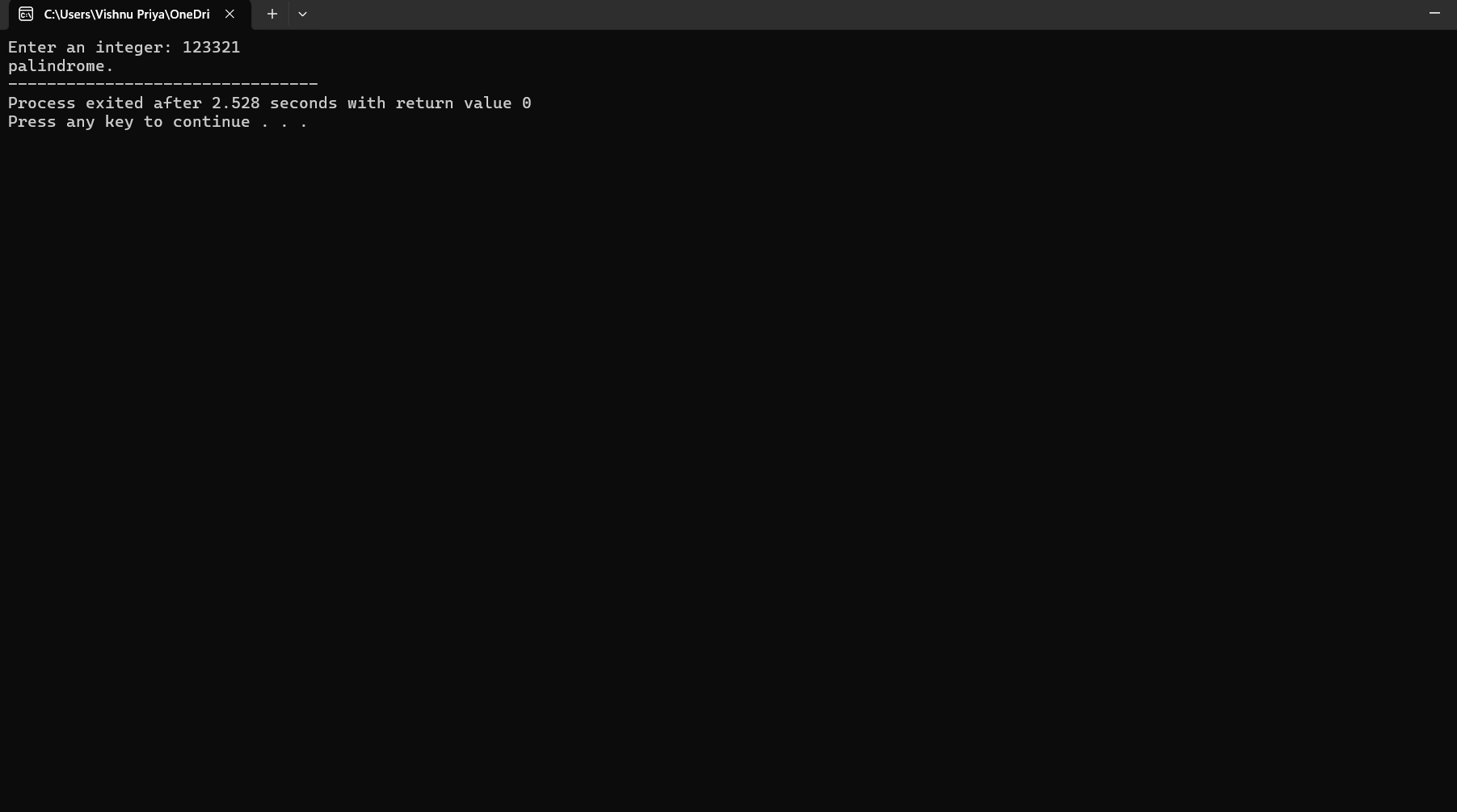
cout<<"palindrome.";

else

cout<<"not a palindrome.";

return 0;

}



6.compare(= =)

#include<iostream>

#include <cstring>

using namespace std;

class String

{

public:

char\* data;

String(const char\* str)

{

int len = strlen(str);

data = new char[len + 1];

strcpy(data,str);

}

};

int main()

{

string str1,str2;

cout<<"enter str1:";

cin>>str1;

cout<<"enter str2:";

cin>>str2;

if (str1 == str2)

{

cout << "str1 and str2 are equal\n";

}

else

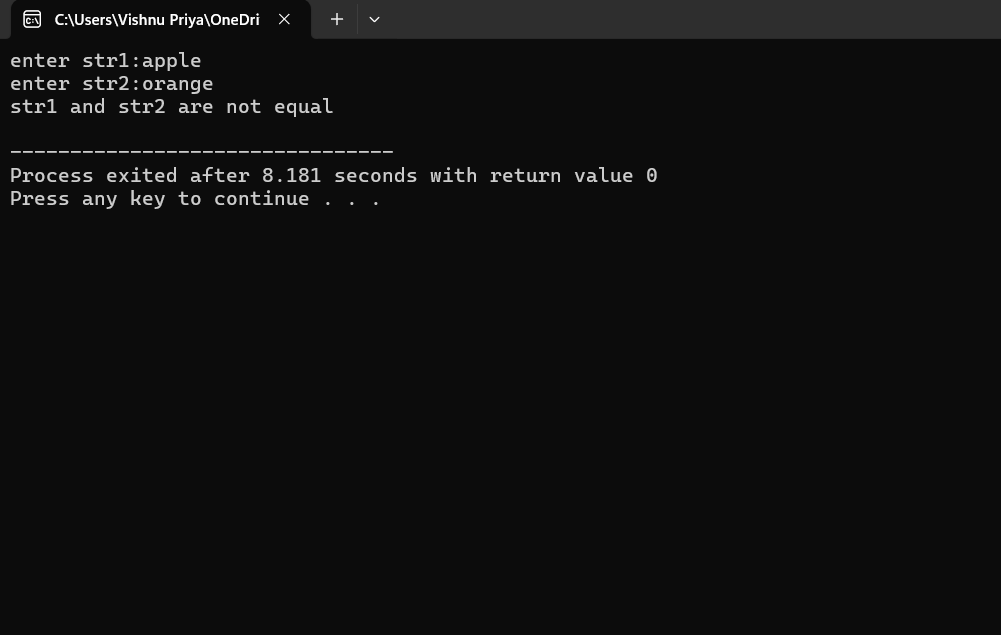
{

cout << "str1 and str2 are not equal\n";

}

return 0;

}



**Medium**

1.add Num end with4

#include <iostream>

using namespace std;

int AddEnd4(int A[][4], int R, int C) {

int sum = 0;

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

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

if (A[i][j] % 10 == 4) {

sum += A[i][j];

}

}

}

return sum;

}

int main() {

int A[][4] = {{24, 16, 14}, {19, 5, 4}};

int R = 2;

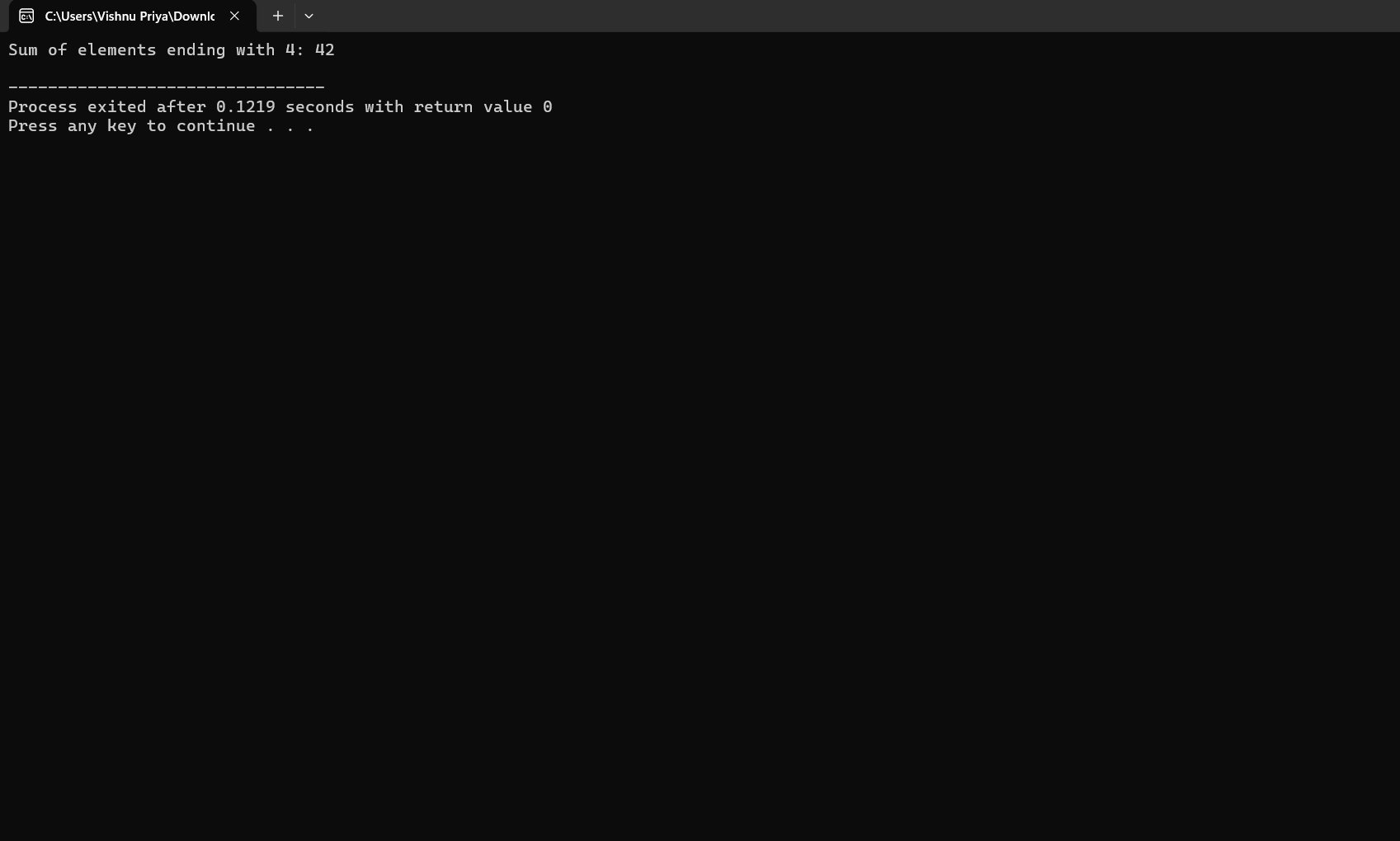
int C = 3;

int sum = AddEnd4(A, R, C);

cout << "Sum of elements ending with 4: " << sum << endl;

return 0;

}



2.reverse arr of ele

#include <iostream>

using namespace std;

void Reverse(int A[], int n)

{

for (int i = 0; i < n / 2; i++)

{

swap(A[i], A[n - i - 1]);

}

}

int main()

{

int A[] = {10, 20, 30, 40, 50};

int n = sizeof(A) / sizeof(A[0]);

cout << "Original array: ";

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

cout << A[i] << " ";

}

cout << endl;

Reverse(A, n);

cout << "Reversed array: ";

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

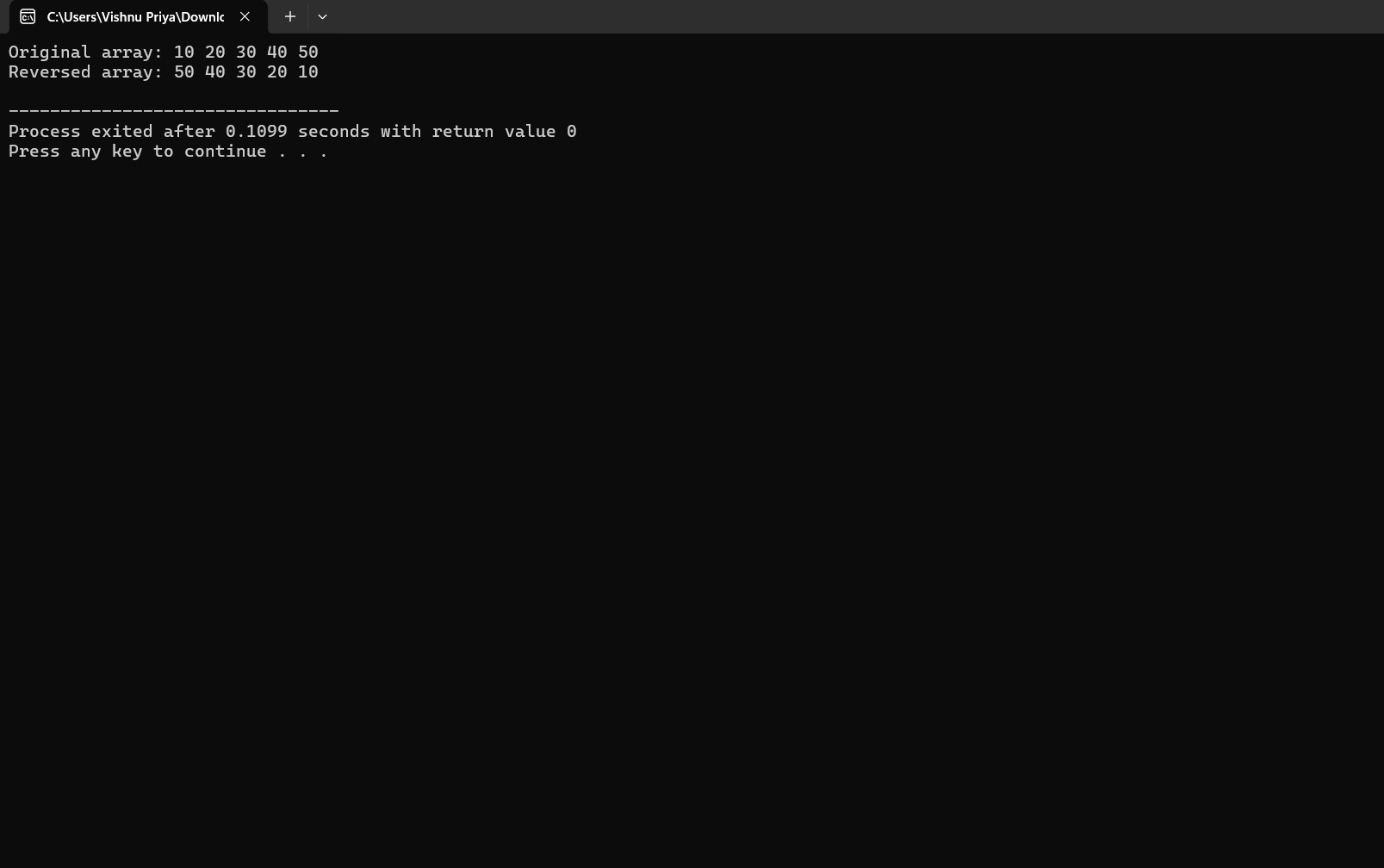
cout << A[i] << " ";

}

cout << endl;

return 0;

}



**Hard**

1.Area of Square,Circle

#include <iostream>

#include <cmath>

using namespace std;

class Shape {

public:

virtual double calculateArea() = 0;

};

class Square : public Shape {

private:

double side;

public:

Square(double s) : side(s) {}

double calculateArea() override {

return side \* side;

}

};

class Circle : public Shape {

private:

double radius;

public:

Circle(double r) : radius(r) {}

double calculateArea() override {

return M\_PI \* radius \* radius;

}

};

int main() {

Square square(4);

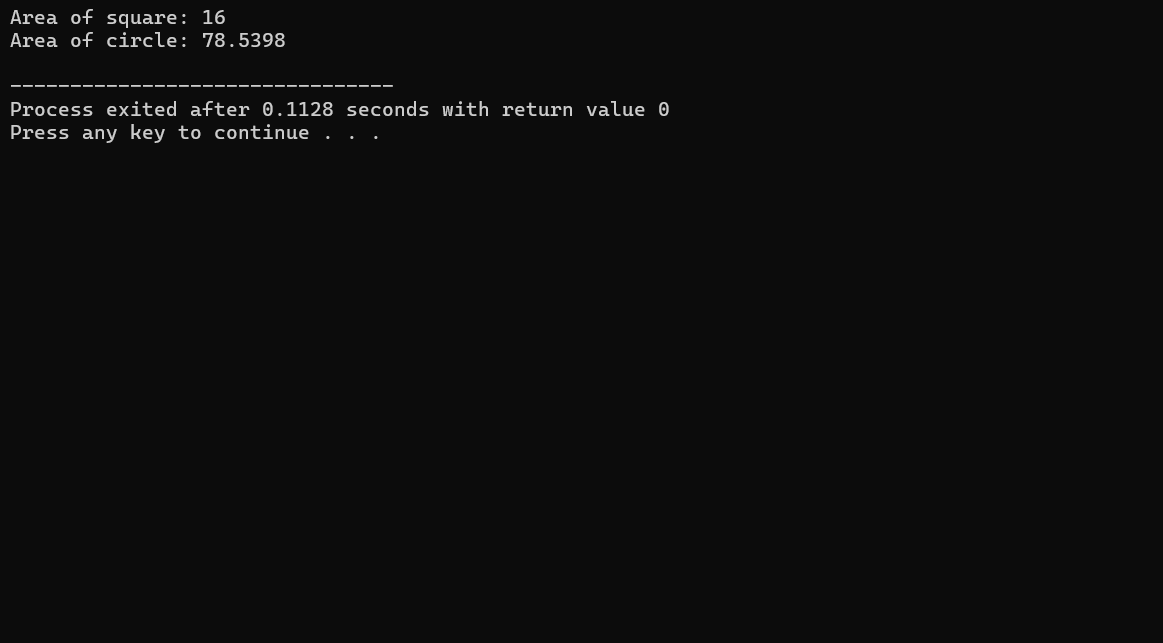
Circle circle(5);

cout << "Area of square: " << square.calculateArea() << endl;

cout << "Area of circle: " << circle.calculateArea() << endl;

return 0;

}



2.No.of word in Paragraph

#include <iostream>

#include <string>

using namespace std;

int countWords(string str)

{

int count = 0;

for (int i = 0; i < str.length(); i++)

{

if (str[i] == ' ')

{

count++;

}

}

return count + 1;

}

int main()

{

string paragraph;

cout << "Enter a paragraph: ";

getline(cin, paragraph);

int wordCount = countWords(paragraph);

cout << "Number of words: " << wordCount << endl;

return 0;

}

