** RIPHAH INTERNATIONAL UNIVERSITY LAHORE**

NAME: **Anousha Ijaz**

SAP: **47544**

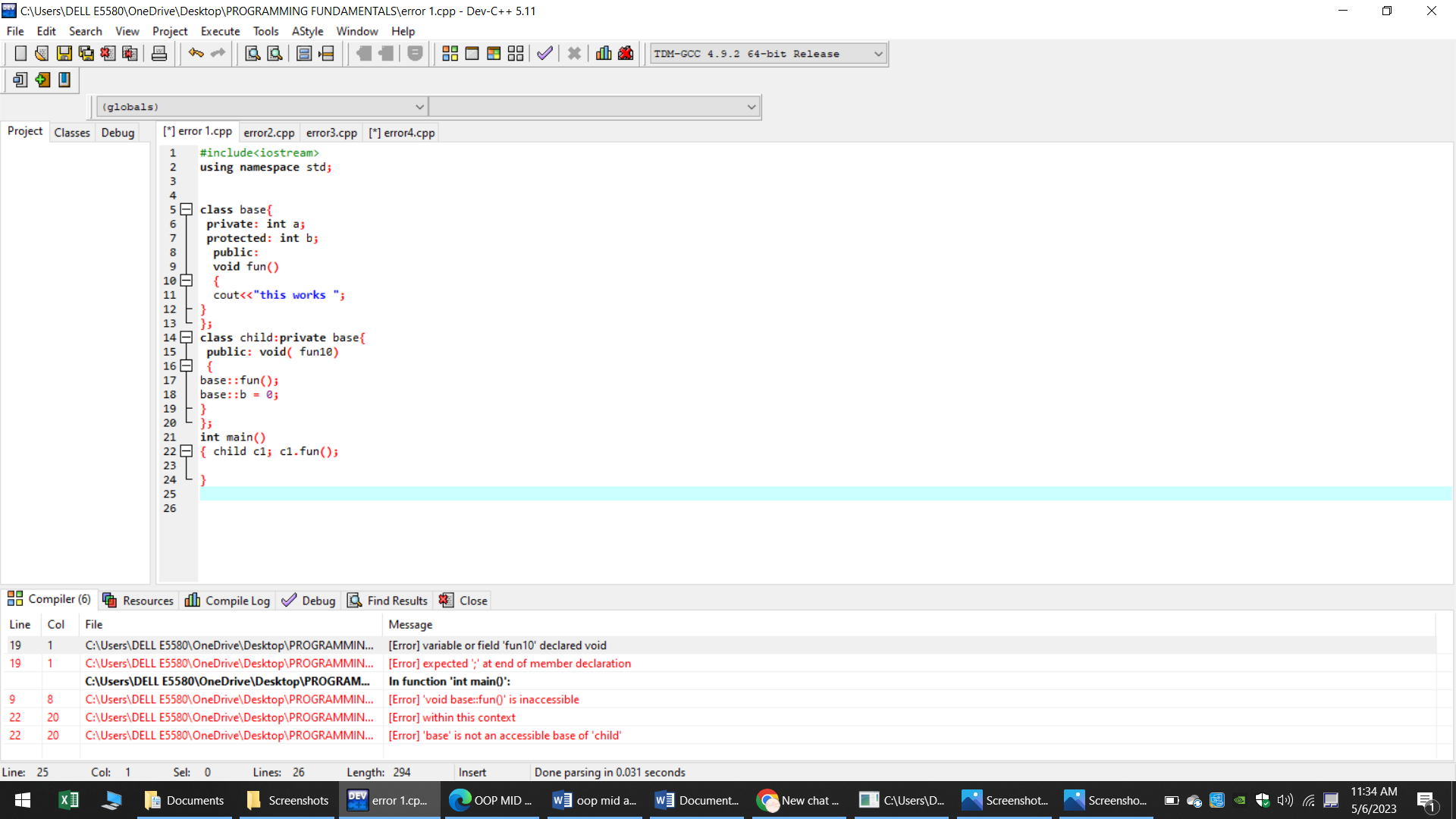
PROGRAM & SEM: **BSCS 2A**

SUBJECT: OOP (object oriented programming)

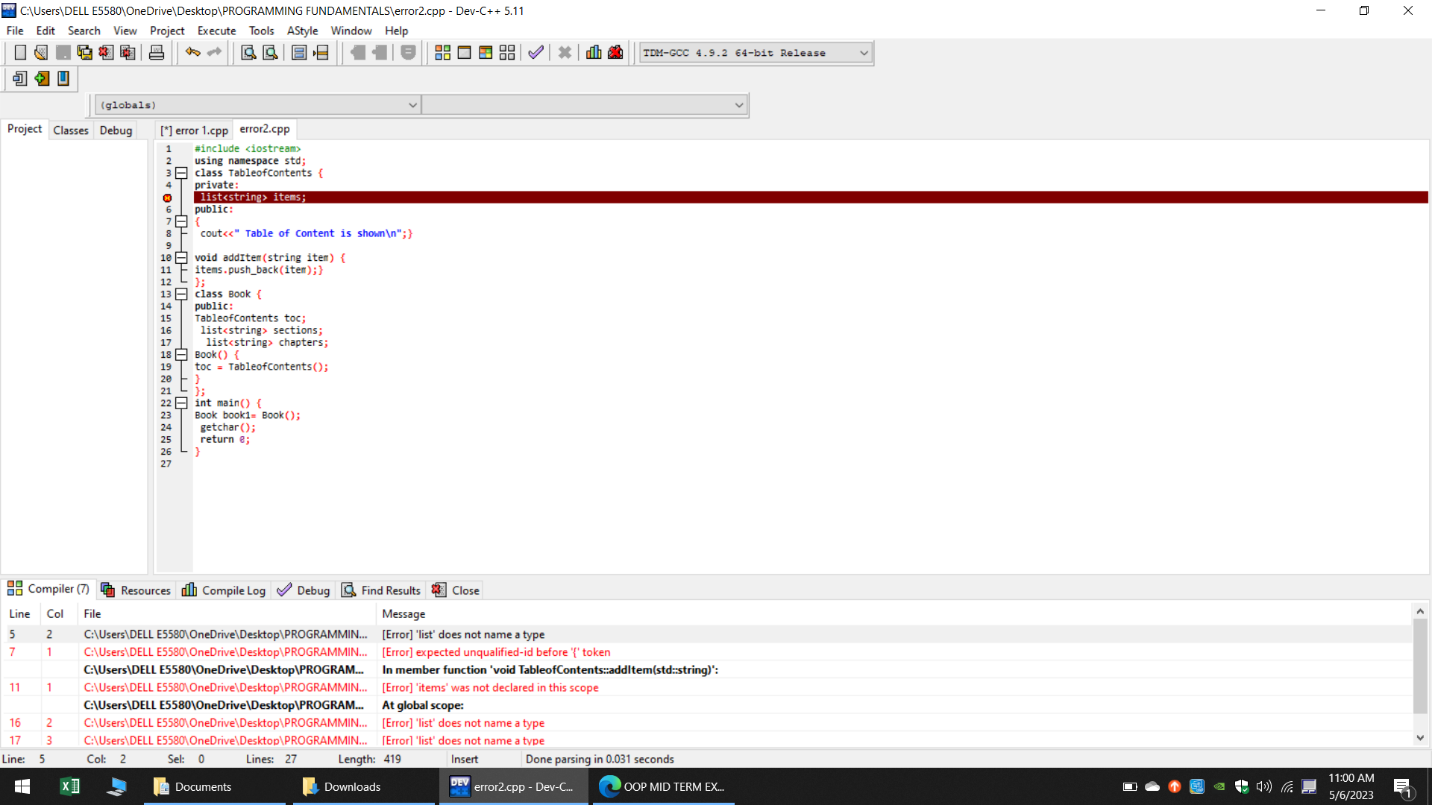
**ASSIGNMENT:**

**OUTPUTS:**

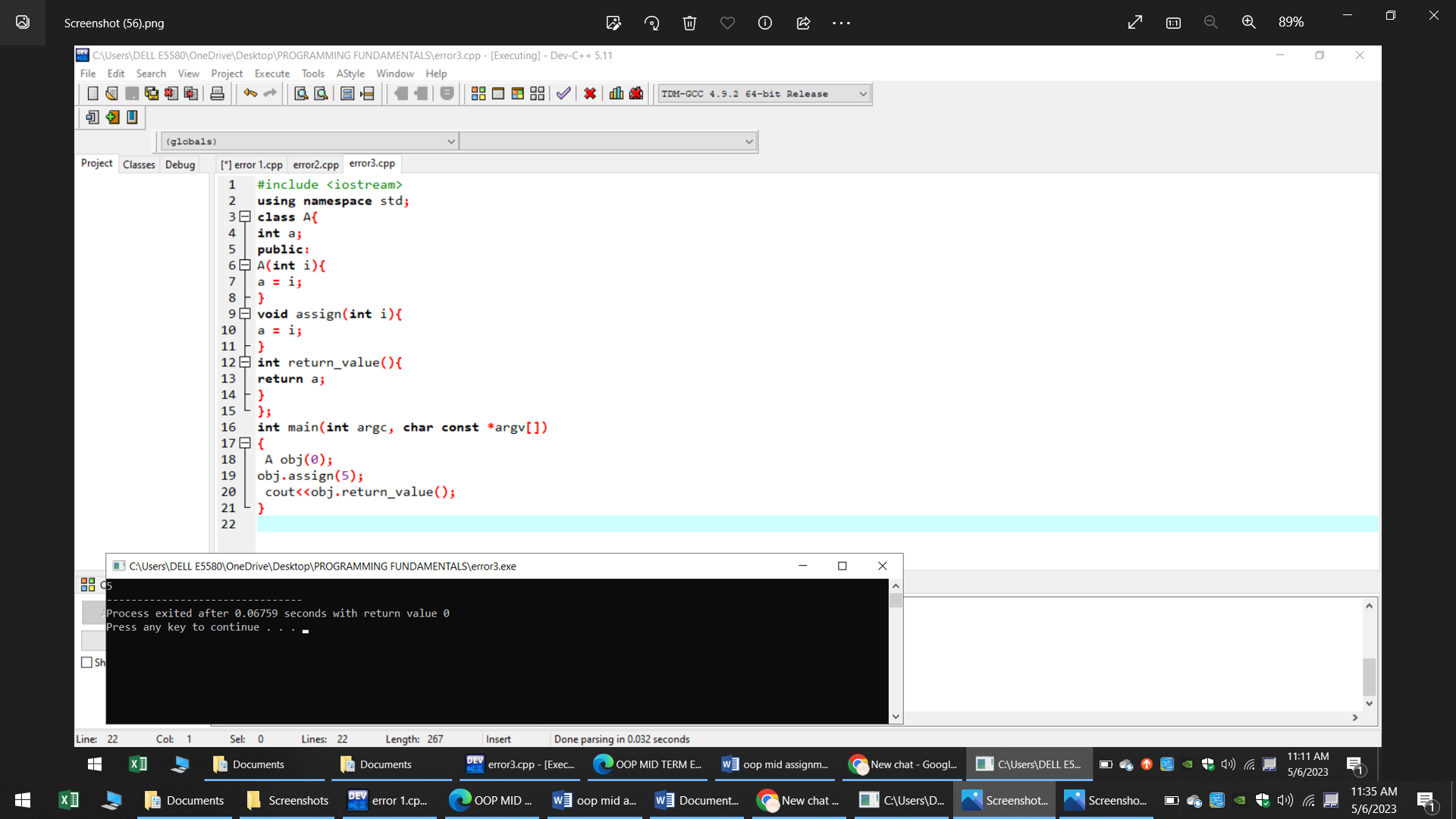
**Q1:**



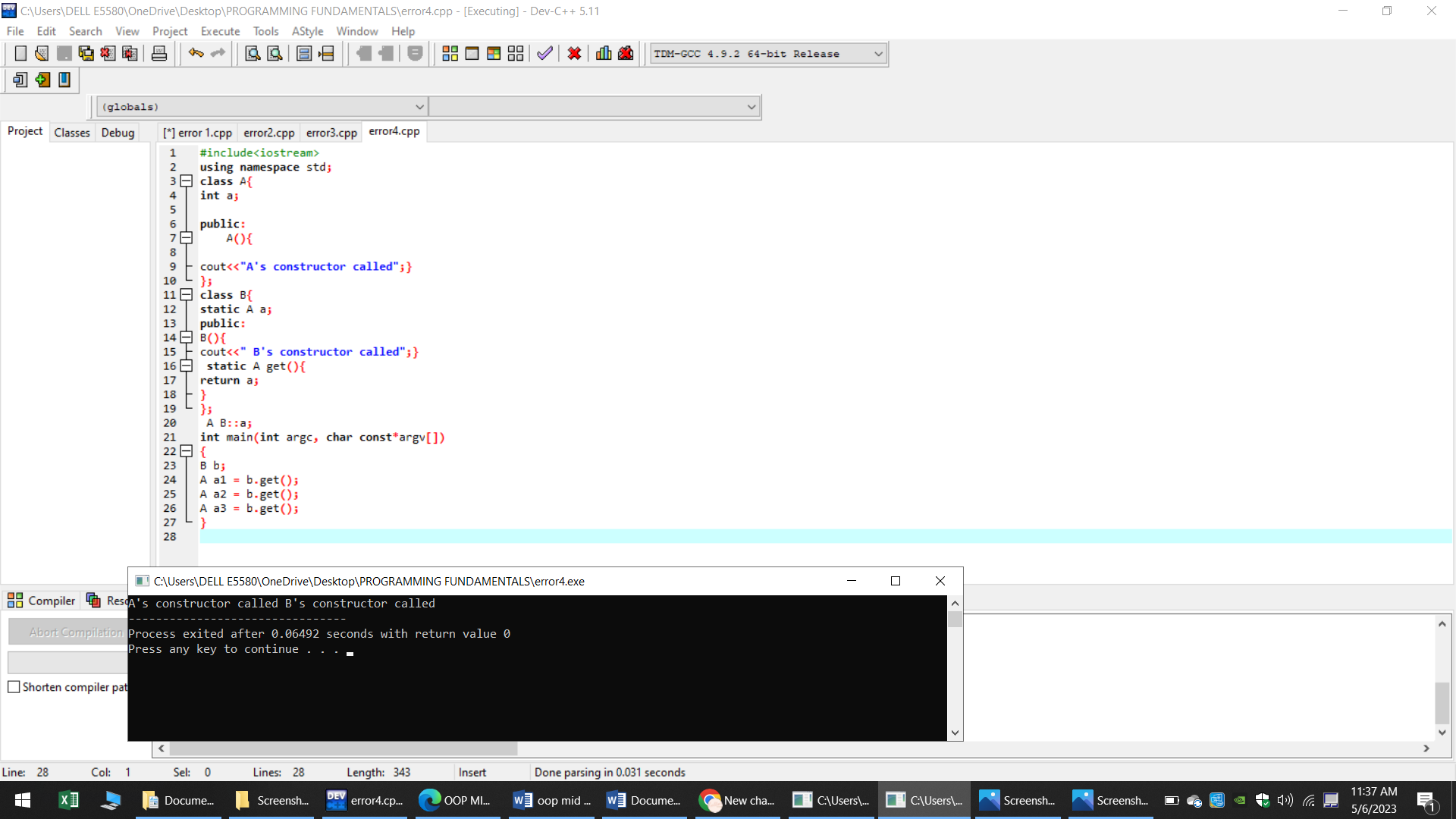
**Q2:**



**Q3:**

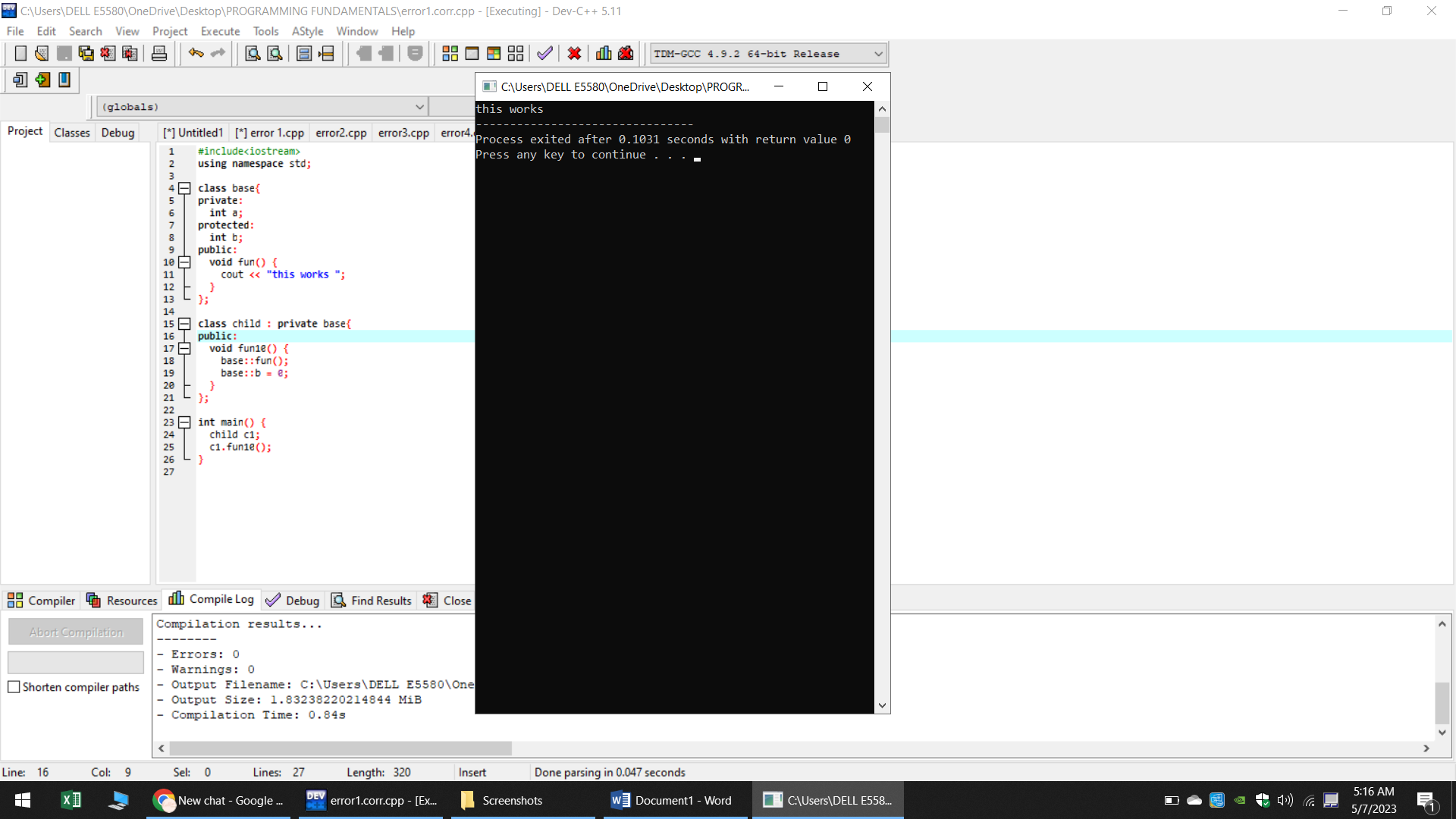


**Q4:**

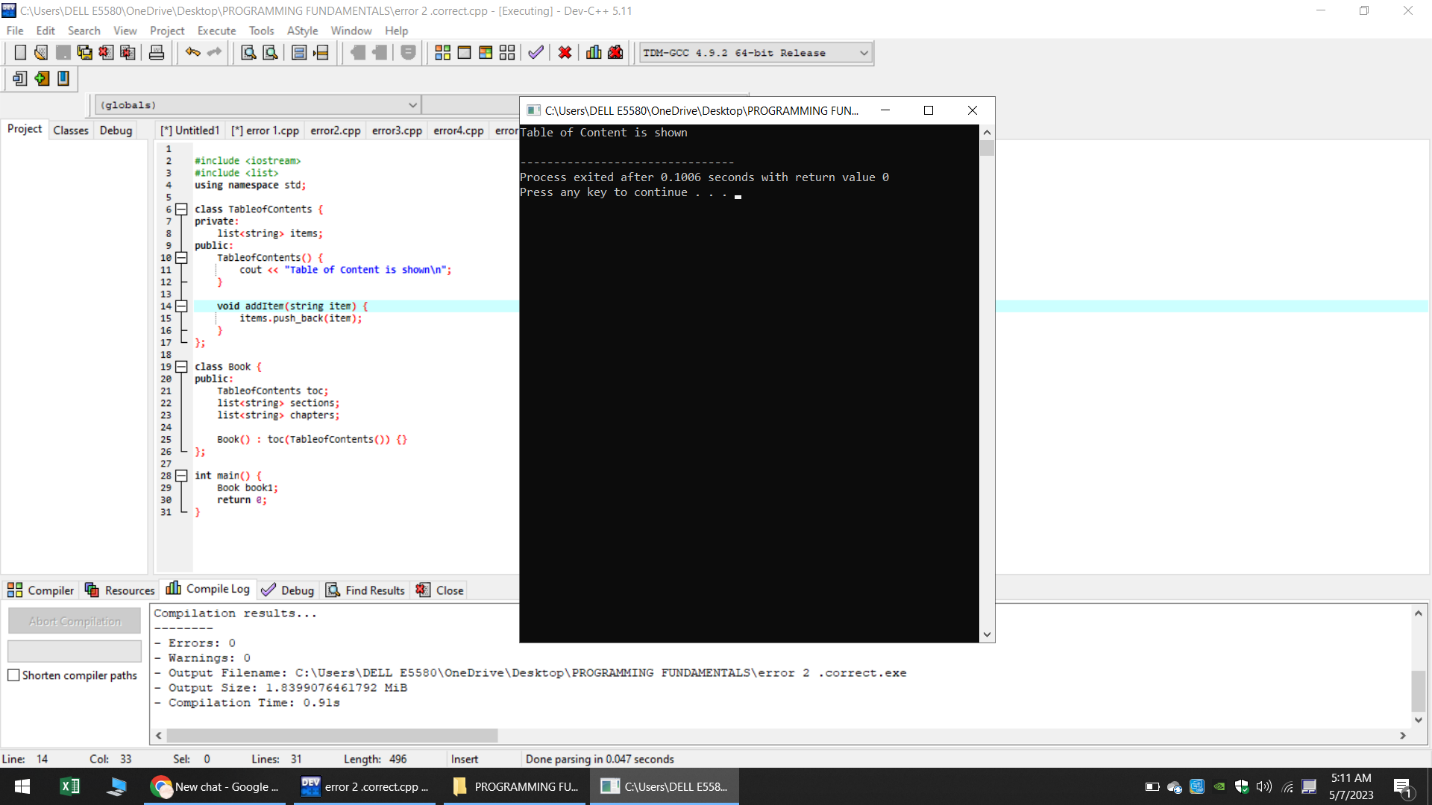


**CORRECTED ERRORS:**

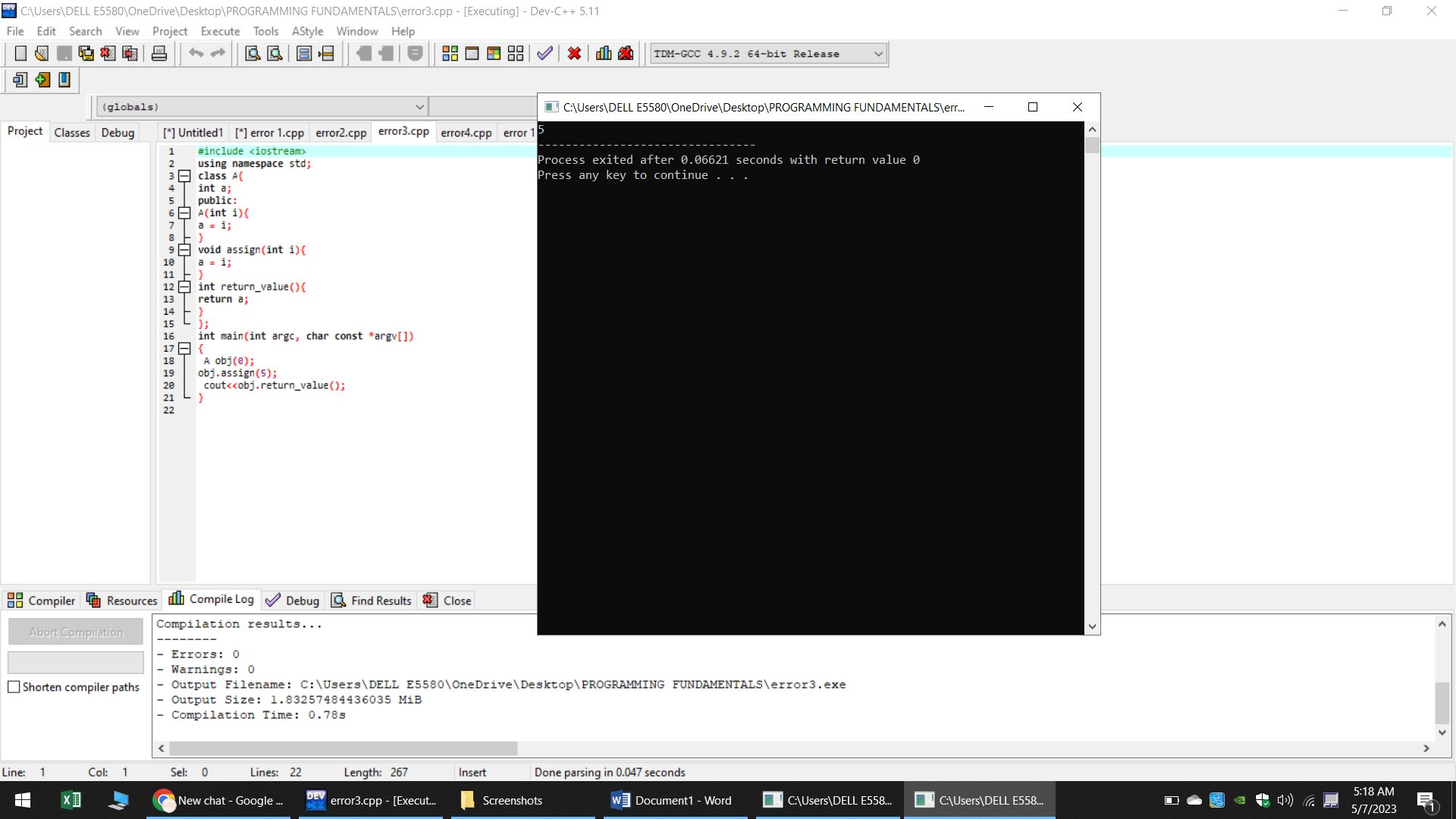
**Q1:**



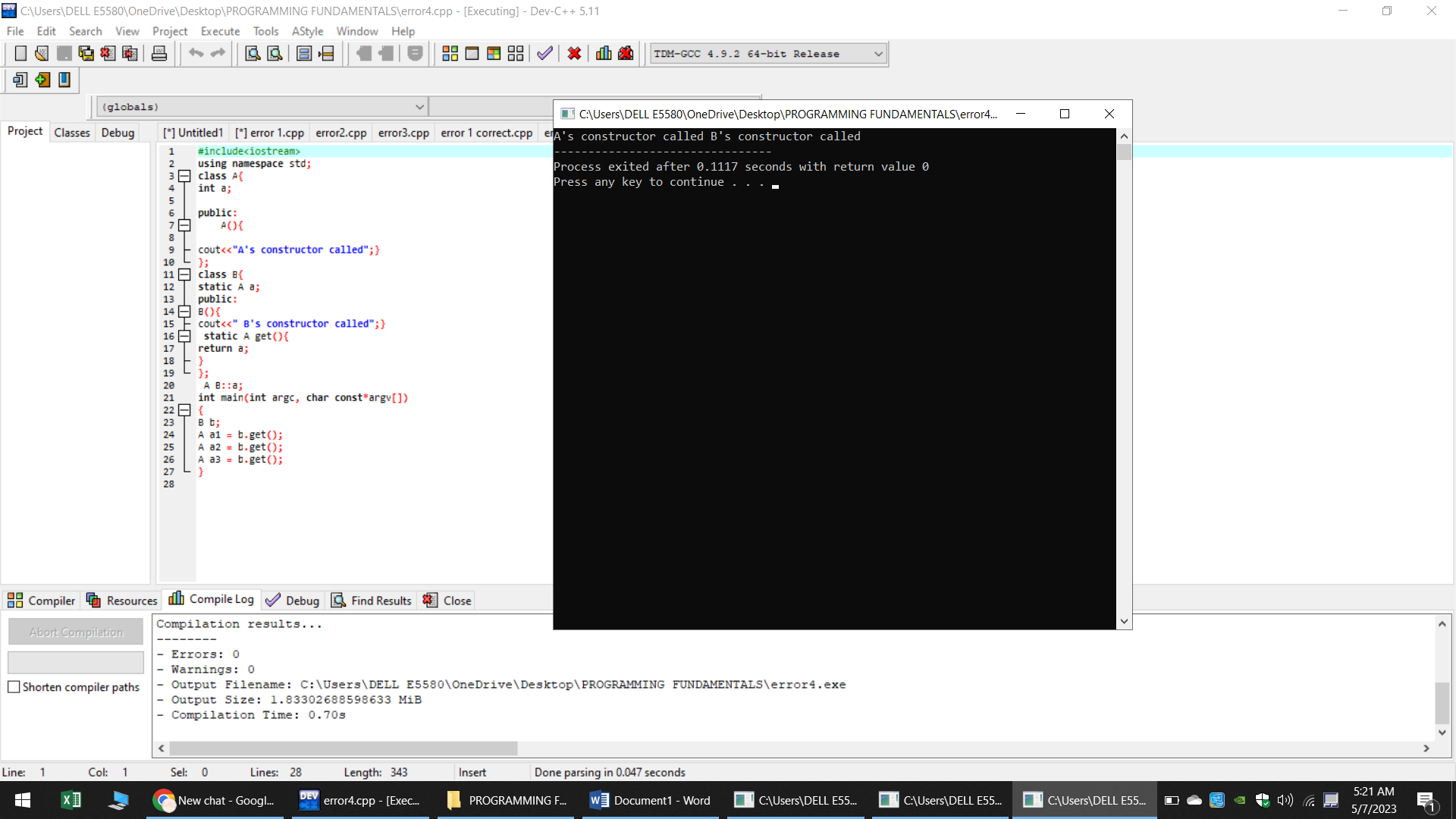
**Q2:**



**Q3:**



**Q4:**



**What is the purpose of access modifiers in programming language c++ ?**

The access modifiers of C++ allow us to determine which class members are accessible to other classes and functions, and which are not.

**If we want to access the private members of the class in the child class what do we need to change in c++?**

If we want to access the private members of the class in the child class we will simply use the public access modifier instead of private access modifier.

**PROGRAM**

#include <iostream>

#include <string>

#include <vector>

using namespace std;

class Character {

protected:

int id;

string name;

int maxPower;

int strength;

public:

Character(int id, string name, int maxPower, int strength)

: id(id), name(name), maxPower(maxPower), strength(strength) {}

void walk() {

cout << name << " is walking." << endl;

}

void jump() {

cout << name << " is jumping." << endl;

}

void eat() {

cout << name << " is eating." << endl;

}

};

class Doremon : public Character {

private:

vector<string> gadgets;

string partnerName;

public:

Doremon(int id, string name, int maxPower, int strength, vector<string> gadgets, string partnerName)

: Character(id, name, maxPower, strength), gadgets(gadgets), partnerName(partnerName) {}

void showGadgets() {

cout << name << " has the following gadgets:" << endl;

for (string gadget : gadgets) {

cout << gadget << endl;

}

}

void launchAttack() {

cout << name << " is launching an attack." << endl;

}

void fly() {

cout << name << " is flying." << endl;

}

};

class Benten : public Character {

private:

string watchName;

vector<string> powers;

int watchCharge;

public:

Benten(int id, string name, int maxPower, int strength, string watchName, vector<string> powers, int watchCharge)

: Character(id, name, maxPower, strength), watchName(watchName), powers(powers), watchCharge(watchCharge) {}

void rotateWatch() {

cout << name << " is rotating the watch." << endl;

}

void fight() {

cout << name << " is fighting." << endl;

}

void drive() {

cout << name << " is driving." << endl;

}

};

int main() {

vector<string> doremonGadgets = {"Anywhere Door", "Take-copter", "Bamboo-copter", "Small Light"};

Doremon doremon(1, "Doremon", 100, 80, doremonGadgets, "Nobita");

doremon.walk();

doremon.jump();

doremon.showGadgets();

doremon.launchAttack();

doremon.fly();

cout << endl;

vector<string> bentenPowers = {"Fire", "Water", "Wind", "Earth"};

Benten benten(2, "Benten", 150, 90, "Omnitrix", bentenPowers, 80);

benten.walk();

benten.jump();

benten.rotateWatch();

benten.fight();

benten.drive();

return 0;

}

**OUTPUT:**