CS 115 Assignment - 2

Question 1.

a) Can be accessed using dot (.) operators for objects as well as a pointer (->) to objects.
b) Difference between public and private access

   Public: can be accessed outside the class through objects and pointers.

   Private: cannot be accessed outside the class and can only be accessed in the class itself.

c) It is used to initialize values, first called when object is created, the name of the constructor is same as of class name and has no return type.

   3 main types:

   1. Default: No parameters/arguments are passed
   2. Parameterized (2 types): Any number of parameters/arguments are passed Implicit Call Explicit Call
   3. Copy: In which we copy one's object to another Mandatory - Copy object on another

Question 2.

Q2.cpp › Student › Student › main

```cpp
#include <iostream>
using namespace std;

class Student {
public:
  string name;
  int roll_no;
  Student() {
    name = "John";
    roll_no = 2;
  }
  void display() {
    cout << "Name: " << name << endl;
    cout << "Roll #: " << roll_no;
  }
};

int main() {
  Student s1;
  s1.display();
  return 0;
}

```

```
g++ Q2.cpp -o q2
./q2
Name: John
Roll #: 2
```

Question 3.

Q3.cpp > ...

```cpp
#include <cmath>
#include <iostream>

using namespace std;

class Traingle {
public:
  double a = 3, b = 5, c = 4, s;

  double calcPerimeter() { return s = (a + b + c); }

  double calcArea() {
    s = (a + b + c) / 2;
    return sqrt(s * (s - a) * (s - b) * (s - c));
  }
};

int main() {
  Traingle T;

  cout << "Perimeter: " << T.calcPerimeter() << endl;
  cout << "Area: " << T.calcArea() << endl;

  return 0;
}

```

```
> g++ Q3.cpp -o q3
./q3
> ./q3
Perimeter: 12
Area: 6
>
```

Question 4.

```cpp
#include <iostream>
#include <string>

using namespace std;

class Car {
private:
  string company;
  string model;
  int year;

public:
  Car(string c, string m, int y) : company(c), model(m), year(y) {}

  string getComp() const { return company; }

  string getModel() const { return model; }

  int getYear() const { return year; }

  void setComp(string c) { company = c; }

  void setModel(string m) { model = m; }

  void setYear(int y) { year = y; }
};

int main() {
  Car TheWhip("Kia", "Soul", 2012);

  cout << "Old car: " << endl;
  cout << "Company: " << TheWhip.getComp() << endl;
  cout << "Model: " << TheWhip.getModel() << endl;
  cout << "Year: " << TheWhip.getYear() << endl;

  TheWhip.setComp("Cadillac");
  TheWhip.setModel("CT4 V BW");
  TheWhip.setYear(2023);

  cout << endl;

  cout << "The new whip: " << endl;
  cout << "Company: " << TheWhip.getComp() << endl;
  cout << "Model: " << TheWhip.getModel() << endl;
  cout << "Year: " << TheWhip.getYear() << endl;

  return 0;
}
```

```
g++ Q4.cpp -o q4
./q4
Old car:
Company: Kia
Model: Soul
Year: 2012

The new whip:
Company: Cadillac
Model: CT4 V BW
Year: 2023
```