**List down the data members and member functions of a class identifying an Android phone. You are allowed to suppose the data members and the member functions of the class. Also create a single member function that will display the entire class data members. You will be graded on the quality and clarity of your design.**

#include <iostream>

using namespace std;

class Andriodphone{

string modelName;

string brand;

int ram;

int storage;

public:

void setModelName(string n){

modelName=n;

}

string getModelName() {

return modelName;

}

void setBrand(string b){

brand=b;

}

string getBrand(){

return brand;

}

void setRam(int r){

ram=r;

}

int getRam(){

return ram;

}

void setStorage(int s){

storage=s;

}

int getStorage(){

return storage;

}

void displayDetails(){

cout<<"Phone details:"<<endl;

cout<<"Model Name:"<<getModelName()<<endl;

cout<<"Brand:"<<getBrand()<<endl;

cout<<"Ram:"<< getRam()<<endl;

cout<<"Storage:"<<getStorage()<<endl;

}

};

int main(){

Andriodphone phone;

phone.setModelName("galaxy s23");

phone.setBrand("samsung");

phone.setRam(8);

phone.setStorage(128);

phone.displayDetails();

return 0;

}

**Identify the data members and member functions for a submarine sandwich class. The class should have all the relevant attributes and qualities required for a sandwich object. Try to be imaginative in your design. For example consider various sauces, Mortadella, salads, cheese and other fillings etc.**

#include <iostream>

using namespace std;

class SubmarineSandwich{

string breadType;

string meatType;

string cheese;

string vegetable;

bool toasted;

double price;

public:

void setBreadType(string b){

breadType=b;

}

string getBreadType(){

return breadType;

}

void setmeatType(string m){

meatType=m;

}

string getmeatType(){

return meatType;

}

void setcheese(string c){

cheese=c;

}

string getcheese(){

return cheese;

}

void setvegetable(string v){

vegetable=v;

price+=1.0;

}

string getvegetable(){

return vegetable;

}

void setToasted(bool t){

toasted=t;

if(toasted){

price+=1.5;

}

}

bool gettoasted(){

return toasted;

}

void setprice(double p){

price=p;

}

double getprice(){

return price;

}

void displayDetails(){

cout<<"Submarine sandwich details:"<<endl;

cout<<"Bread Type:"<<getBreadType()<<endl;

cout<<"Meat Type:"<<getmeatType()<<endl;

cout<<"Cheese:"<<getcheese()<<endl;

cout<<"Vegetable:"<< getvegetable()<<endl;

cout << "Toasted: " << (isToasted() ? "Yes" : "No") << endl;

cout<<"Price:"<< getprice()<<endl;

}

};

int main(){

SubmarineSandwich mysandwich;

mysandwich.setBreadType("Whole Wheat");

mysandwich.setmeatType("Turkey");

mysandwich.setcheese("Cheddar");

mysandwich.setvegetable("Lettuce,Tomato");

mysandwich.setToasted(true);

mysandwich.setprice(5000);

mysandwich.displayDetails();

return 0;

}

**Write a program that creates a class called arc. The data members of the class are radius, angle and arc\_length of the arc is provided. Write three functions that set the respective values, and then write a single constant function that will read the values**

#include<iostream>

#include<cmath>

using namespace std;

class Arc{

private:

double radius;

double angle;

double arc\_length;

public:

void setRadius(double r){

radius=r;

}

void setAngle(double a){

angle=a;

}

void setArc\_length(double arc){

arc\_length=arc;

}

void readValues() const{

cout<<"Radius:"<<radius<<endl;

cout<<"Angle:"<<angle<<endl;

cout<<"Arc Length"<<arc\_length<<endl;

}

void calculateArcLength(){

arc\_length=(angle/360)\*2\*M\_PI\*radius;

}

};

int main(){

Arc myArc;

myArc.setRadius(5.0);

myArc.setAngle(90.0);

myArc.calculateArcLength();

myArc.readValues();

return 0;

}

**Create a class Android\_Device. The data members of the class are IMEIno (int), Type (String), Make (String), Modelno (int), Memory(float), Operating\_System(String). Then Implement member functions to: 1. Set the values of all data members. 2. Display the values of all data members**

#include <iostream>

#include <string>

using namespace std;

class Android\_Device {

private:

int IMEIno;

string Type;

string Make;

int Modelno;

float Memory;

string Operating\_System;

public:

void setValues(int imei, string t, string m, int mNO, float mem, string os) {

IMEIno = imei;

Type = t;

Make = m;

Modelno = mNO;

Memory = mem;

Operating\_System = os;

}

void displayValues() {

cout << "IMEI Number: " << IMEIno << endl;

cout << "Type: " << Type << endl;

cout << "Make: " << Make << endl;

cout << "Model Number: " << Modelno << endl;

cout << "Memory: " << Memory << " GB" << endl;

cout << "Operating System: " << Operating\_System << endl;

}

};

int main() {

Android\_Device myDevice;

myDevice.setValues(123456789, "Smartphone", "Samsung", 101, 64.5, "Android");

myDevice.displayValues();

return 0;

}

**Write a class called quadrilateral. Your task is to store the length of all the sides of the quadrilateral and the value of the 2 opposite angles within the quadrilateral. Then implement memberfunctions: 1. To compute the area of the quadrilateral. 2. To compute the parameter of the quadrilateral. 3. A constant function that will display the length of all the sides, the angles, the parameter of the quadrilateral and area of the quadrilateral. 4. Create setter and getter methods. Demonstrate the use of**

#include<iostream>

#include<cmath>

using namespace std;

class Quadrilateral {

private:

double side1,side2,side3,side4;

double angle1,angle2;

public:

void setSides(double s1, double s2, double s3,double s4){

side1=s1;

side2=s2;

side3=s3;

side4=s4;

}

void setAngles(double a1, double a2){

angle1=a1;

angle2=a2;

}

double getPerimeter(){

return side1+side2+side3+side4;

}

double getArea(){

double s=getPerimeter()/2;

return sqrt(s\*(s-side1)\*(s-side2)\*(s-side3)\*(s-side4));

}

void display (){

cout<<"Sides:"<<side1<<","<<side2<<","<<side3<<","<<side4<<endl;

cout<<"Angle:"<<angle1<<","<<angle2<<endl;

cout<<"Perimeter:"<<getPerimeter()<<endl;

cout<<"Area:"<<getArea()<<endl;

}

};

int main(){

Quadrilateral quad;

quad.setSides(5.0,6.0,6.0,9.0);

quad.setAngles(90.0,90.0);

quad.display();

return 0;

}

**Your task is to carefully understand the code provided. Analyze the code and suggest any corrections that may be needed. There are both syntax and logical errors in the code so consider both when designing the correct solution. Submit the correct code to the lab instructor. class student{ int age; int cnic; int semester; char name; public: int setall(int a, int c, int s, int sem, char n) const { age = a; cnic = c; semester = s; name = n; } } int myclass :: displayall ( ){ cout<<**

#include <iostream>

#include <string>

using namespace std;

class Student {

int age;

int cnic;

int semester;

string name;

public:

void setAll(int a, int c, int s, int sem, string n) {

age = a;

cnic = c;

semester = sem;

name = n;

}

void displayAll() const {

cout << "The entered data is: " << endl;

cout << "Name: " << name << endl;

cout << "Age: " << age << endl;

cout << "CNIC: " << cnic << endl;

cout << "Semester: " << semester << endl;

}

};

int main() {

Student obj;

obj.setAll(20, 123456, 3, 4, "Aisha");

obj.displayAll();

Student anotherObj;

anotherObj.setAll(22, 78910, 5, 6, "Jami");

anotherObj.displayAll();

return 0;

}

**Create a class “Vehicle” having: ● “LisencePlate\_No”, “Model\_No”, “Type”, and “Color” as private data members. ● Parameterized constructor assigning default values to the data members ● No-Argument constructor assigning default values to the data members ● Default constructor assigning default values to the data members ● Public member functions: Register Vehicle, Update Vehicle, Delete Vehicle, and Search Vehicle NOTE: The values required should be obtained from the user in “void main ()” function and each data member should have a setter and a getter through which you are** required **to perform the memberfunction**