**Test 3**

Yuyang Zhang

**Q1:**

#include<iostream>

#include <cstring>

using namespace std;

class Rectangle

{

private:

float length;

float width;

public:

void setlength(float l)

{

length = l;

}

void setwidth(float w)

{

width = w;

}

float perimeter()

{

return 2 \* (length + width);

}

float area()

{

return length \* width;

}

void show()

{

cout<<"Rectangle length is: "<<length<<endl;

cout<<"Rectangle width is: "<<width<<endl;

cout<<"Rectangle area is: "<<area()<<endl;

cout<<"Rectangle perimeter is: "<<perimeter()<<endl;

}

int sameArea(Rectangle r)

{

if(area() == r.area())

return 1;

return 0;

}

};

int main()

{

Rectangle r1;

r1.setlength(5);

r1.setwidth(2.5);

r1.show();

Rectangle r2;

r2.setlength(5);

r2.setwidth(18.9);

r2.show();

if(r1.sameArea(r2))

cout<<"r1 has same area with r2\n";

else

cout<<"r1 has different area with r2\n";

r1.setlength(15);

r1.setwidth(6.3);

if(r1.sameArea(r2))

cout<<"r1 has same area with r2\n"<<endl;

else

cout<<"r1 has different area with r2\n"<<endl;

return 0;

}

**Q2:**

#include<iostream>

#include <cstring>

using namespace std;

class Complex

{

private:

float real;

float imag;

public:

void setN(float r, float i)

{

real = r;

imag = i;

}

void disp()

{

cout<<"real number is: "<<real<<endl;

cout<<"imaginary number is: "<<imag<<endl;

}

Complex sum(Complex a)

{

Complex n;

n.real = real + a.real;

n.imag = imag + a.imag;

return n;

}

};

int main()

{

Complex c1;

Complex c2;

Complex c3;

c1.setN(1.1, 2.2);

c2.setN(3.3, 4.4);

c3 = c1.sum(c2);

c1.disp();

c2.disp();

c3.disp();

return 0;

}

**Q3:**

#include<iostream>

#include <cstring>

using namespace std;

class Distance

{

private:

int feet;

float inches;

public:

void setD(int f, float i)

{

feet = f;

inches = i;

}

void disp()

{

cout<<"Feet is: "<<feet<<endl;

cout<<"Inches is: "<<inches<<endl;

}

Distance add(Distance d)

{

Distance n;

n.feet = feet + d.feet;

n.inches = inches + d.inches;

return n;

}

};

int main()

{

Distance d1;

Distance d2;

Distance d3;

d1.setD(1, 2.2);

d2.setD(3, 4.4);

d3 = d1.add(d2);

d1.disp();

d2.disp();

d3.disp();

return 0;

}

**Q4:**

#include<iostream>

#include <cstring>

using namespace std;

class Time

{

private:

int hours;

int minutes;

public:

void settime(int h, int m)

{

hours = h;

minutes = m;

}

void showtime()

{

cout<<"Hours is: "<<hours<<endl;

cout<<"Minutes is: "<<minutes<<endl;

}

Time sum(Time t)

{

Time n;

n.hours = hours + t.hours;

n.minutes = minutes + t.minutes;

return n;

}

};

int main()

{

Time t1;

Time t2;

Time t3;

t1.settime(1, 27);

t2.settime(3, 31);

t3 = t1.sum(t2);

t1.showtime();

t2.showtime();

t3.showtime();

return 0;

}

**Q5:**

#include<iostream>

#include <cstring>

using namespace std;

class cashRegister

{

private:

int cashOnHand;

public:

cashRegister()

{

cashOnHand = 500;

}

cashRegister(int cash)

{

cashOnHand = cash;

}

int getCurrentBalance()

{

return cashOnHand;

}

void acceptAmount(int amount)

{

cashOnHand = cashOnHand + amount;

}

};

class dispenserType

{

private:

int numberOfItems;

int cost;

public:

dispenserType()

{

numberOfItems = 50;

cost = 50;

}

dispenserType(int n, int c)

{

numberOfItems = n;

cost = c;

}

int getNoOfItems()

{

return numberOfItems;

}

int getCost()

{

return cost;

}

void makeSale()

{

numberOfItems--;

}

};

void showSelection();

void sellProduct(dispenserType& product, cashRegister& pCounter);

int main()

{

cashRegister cr;

dispenserType candy(80, 20);

dispenserType chips(60, 60);

dispenserType gum(50, 30);

dispenserType cookies(40, 100);

int choice;

showSelection();

cin >> choice;

while (choice != 5)

{

switch (choice)

{

case 1:

sellProduct(candy, cr);

break;

case 2:

sellProduct(chips, cr);

break;

case 3:

sellProduct(gum, cr);

break;

case 4:

sellProduct(cookies, cr);

break;

default:

cout << "Invalid choice." << endl;

}

showSelection();

cin >> choice;

}

return 0;

}

void showSelection()

{

cout << "Welcome to The World Best Super Candy Shop!" << endl;

cout << "Enter a number to select your prefer snack" << endl;

cout << "Enter 1 for Candy" << endl;

cout << "Enter 2 for Chips" << endl;

cout << "Enter 3 for Gum" << endl;

cout << "Enter 4 for Cookies" << endl;

cout << "Enter 5 to exit" << endl;

}

void sellProduct(dispenserType &item, cashRegister &pcr)

{

int amount;

int extra;

if (item.getNoOfItems() > 0) //if the dispenser is not empty

{

cout << "Deposit " << item.getCost() << " cents" << endl;

cin >> amount;

if (amount < item.getCost())

{

cout << "Still neeed deposit " << item.getCost() - amount << " cents" << endl;

cin >> extra;

amount = amount + extra;

}

if (amount >= item.getCost())

{

pcr.acceptAmount(amount);

item.makeSale();

cout << "Thank you for your shopping today!"<< endl;

}

else

cout << "Your deposit is not enough! " << "Please collect your deposited items." << endl;

}

else

cout << "Sorry, item is sold out!" << endl;

}