[ Ravi Patel ] Instructor: Dr. Hindo

CPSC 230

Chapter 10 - lab assignment (15 pts.)

Drop in assignment 10 dropbox

**Q1:**

Construct a structure that represents a bank certificate of deposit (often called CD). A CD naturally has three pieces of data associated with it: The account, interest rate and the term. Write a main program to get the account information for three customers using get\_data() function. Then write a function calculate\_balance to return the new calculated balance.

struct CDAccount

{

double balance;

double interest\_rate;

int term; //months until maturity

};

void get\_data(CDAccount& the\_account); // get the\_account.balance and

// the\_account.interest\_rate and the\_account.term from the keyboard.

double calculate\_balance (CDAccount& the\_account); // calculate\_balance to return the new calculated balance.

//CPSC 230 RAVI PATEL Assignment 10 Question 1 - structs/bank certificate of deposit

#include <iostream>

using namespace std;

struct CDAccount

{

double balance;

double interest\_rate;

int term; //months until maturity

};

CDAccount account; //declare account as type CDAccount

double rate, interest; //declare rate, interest as double variables

void get\_data(CDAccount& the\_account); // get the\_account.balance and

// the\_account.interest\_rate and the\_account.term from the keyboard.

double calculate\_balance (CDAccount& the\_account); // calculate\_balance to return the new calculated balance.

void display(); //display the results of the program run

void get\_data(CDAccount& the\_account){

cout << "What is the account balance?: $";

cin >> the\_account.balance;

cout << "What is the interest rate of the account?: ";

cin >> the\_account.interest\_rate;

cout << "What is the account term? (How many months until CD maturity?): ";

cin >> the\_account.term;

}

double calculate\_balance (CDAccount& the\_account){

rate = account.interest\_rate / 100.0; //the interest rate is the percentage rate of interest (so we divide by 100)

interest = account.balance \* rate \* (account.term / 12.0); //interest is the current acc balance \* interest rate \*

//times the (terms until maturity / 12 for the whole year - 12 months in a year)

account.balance = account.balance + interest; //new account balance is the old balance plus the interest of account

return 0;

}

void display(){

cout<<"When the account matures, at the end of "<<account.term<<" months, the balance will be a total of: $"<<account.balance<<endl;

cout<<"Thank you for using Ravi's CDAccount Calculator!";

}

int main(int argc, char \*argv[]) {

get\_data(account); //get data for account from user

calculate\_balance(account); //calculate using the given account data

display();

}

**SAMPLE OUTPUT:**

**What is the account balance?: $40000**

**What is the interest rate of the account?: 85**

**What is the account term? (How many months until CD maturity?): 6**

**When the account matures, at the end of 6 months, the balance will be a total of: $57000**

**Thank you for using Ravi's CDAccount Calculator!**

**Q2:**

Run the program in slide 13 to verify the access rule for private and private members: data1, data2” and public member “data 3”. Show in a test program how to access data1, data2 and data3?

//in the program, data1 of o2 and data2 of o1 are not used

//You can access data1, data2, and data3 using the dot operator

//CPSC 230 RAVI PATEL Assignment 10 Question 2

#include <iostream>

using namespace std;

class Test

{

private:

int data1;

float data2;

public:

int data3;

void insertIntegerData(int d) {

data1 = d;

cout << "Number: " << data1;

}void insertIntegerData\_2(int e) {

data2 = e;

cout << "\nNumber: " << data2;

}void insertIntegerData\_3(int f) {

data3 = f;

cout << "\nNumber: " << data3;

}

float insertFloatData() {

cout << "\nEnter data: ";

cin >> data2;

return data2;

}

};

int main()

{

Test o1, o2;

float secondDataOfObject2;

o1.insertIntegerData(12); //**access data1 using dot operator**

o1.insertIntegerData\_2(13); //**access data2 using dot operator**

o1.insertIntegerData\_3(14); //**access data3 using dot operator**

secondDataOfObject2 = o2.insertFloatData();

cout << "You entered " << secondDataOfObject2;

return 0;

}

**SAMPLE OUTPUT:**

**Number: 12**

**Number: 13**

**Number: 14**

**Enter data: 23.2**

**You entered 23.2**

**Q3:**

Write a class box that has

1. private member: length, width and the height
2. Get function to input the private members
3. Public functions volume ( ), total\_area ( ) to to return volume and area of the box.

//CPSC 230 RAVI PATEL Assignment 10 Question 3

#include <iostream>

using namespace std;

double length = 0;

double width = 0;

double height = 0;

double volume = 0;

double area = 0;

class Box{

public:

void input();

void output();

void get\_volume();

//returns the volume of the box

void total\_area();

//returns the total area of the box

double get\_length();

//returns the length of the box

double get\_width();

//returns the width of the box

double get\_height();

//returns the height of the box

private:

double box\_length();

double box\_width();

double box\_height();

};

int main(int argc, char \*argv[]) {

Box cube; //declare cube as type Box

cube.input(); //get user input

cube.get\_volume(); //calculate volume using given input

cube.total\_area(); //calculate total area using given input

cube.output(); //display output to the screen

return 0;

}

void Box::input(){

cout << "Enter the length of the box (inch): ";

cin >> length;

cout << "Enter the width of the box (inch): ";

cin >> width;

cout << "Enter the height of the box (inch): ";

cin >> height;

}

void Box::output(){

cout << "\nLength = " << length << " inches"

<< "\nWidth = " << width << " inches"

<< "\nHeight = " << height << " inches"

<< "\nVolume = " << volume << " cubic inches"

<< "\nTotal Area = " << area << " squared inches" << endl;

}

double Box::get\_length(){

return length;

}

double Box::get\_width(){

return width;

}

double Box::get\_height(){

return height;

}

void Box::get\_volume(){

volume = length\*width\*height; //volume calculation

}

void Box::total\_area(){

area = (2\*(height \* width)) + (2\*(height \* length)) + (2\*(width \* length)); //area calculation

}

**SAMPLE OUTPUT:**

**Enter the length of the box (inch): 10**

**Enter the width of the box (inch): 10**

**Enter the height of the box (inch): 10**

**Length = 10 inches**

**Width = 10 inches**

**Height = 10 inches**

**Volume = 1000 cubic inches**

**Total Area = 600 squared inches**