Trent Giever

Chapter Extra Credit

6/12/18

Programming Challenge ch 5 problem # 23

Code:

//Extra Xredit program

//Ch 5 problem 23

#include <iostream>

#include <iomanip>

#include <math.h>

using namespace std;

int main()

{

double interestRate=0,interestEarned =0, endBal=0, startBal=0 ,deposts=0, withdraws=0;

cout << "Bank program" << endl << endl;

cout << "What is the starting balance: ";

cin >> startBal;

endBal = startBal;

cout << "Please enter the interest rate percentage in decimal form: ";

cin >> interestRate;

for(int i =1; i < 4; i++)

{

double temp1,temp2;

cout << "Please enter the amount deposited in the account during the "<< i << " month: ";

cin >> temp1;

cout << "Please enter the amount withdrawn from the account during the " << i << " month: ";

cin >>temp2;

if(temp1 >= 0)

{

deposts += temp1;

endBal +=temp1;

}

if(temp2 >= 0)

{

if(temp2 <= endBal)

{

withdraws += temp2;

endBal -=temp2;

}

}

interestEarned += (endBal \* (1 + (interestRate/12)))-endBal;

endBal+=interestEarned;

//cout << interestEarned;

}

cout << fixed << showpoint << setprecision(2);

cout << "The start balance is: $ " << startBal <<endl;

cout << "The deposits total is: $ " << deposts << endl;

cout << "The withdraws total is: $ " << withdraws<<endl;

cout << "The interest total is: $ " << interestEarned<<endl;

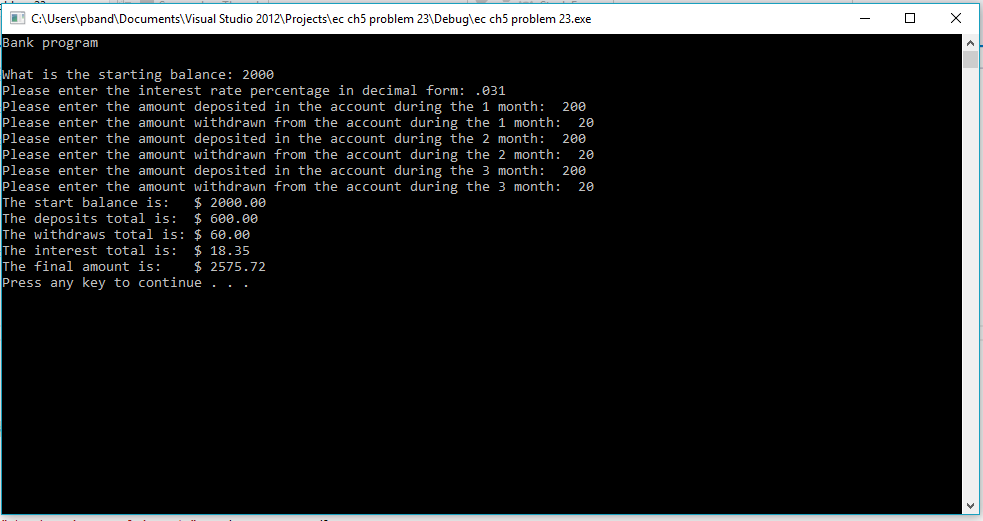
cout << "The final amount is: $ " << endBal<<endl;

system("pause");

return 0;

}

Output:



Programming Challenge ch 6 problem 13

Code:

#include <iostream>

using namespace std;

void getOrderInfo(int &numOrdered, int &numInStock, double &specialShipping)

{

cout << "How many spools in stock: ";

cin >> numInStock;

cout << "How many spools ordered: ";

cin >> numOrdered;

cout << "Any extra shipping fees: ";

cin >> specialShipping;

}

void displayOrderInfo(int numOrdered, int numInStock, double specialShipping, double COST\_PER\_SPOOL, double shippingCost)

{ int temp = 0;

if(numOrdered > numInStock)

{

cout <<endl << "The amount on back order is: " << (numOrdered-numInStock) << endl;

cout << "The total amount in stock ready to ship is: " << (numInStock) << endl;

temp = numInStock;

}

else

{

cout << "The total amount in stock ready to ship is: " << (numOrdered) << endl;

temp = numOrdered;

}

cout << "Total selling price of the stock is: $" << (temp\*COST\_PER\_SPOOL) <<endl;

cout << "Total shipping and handling is: $" << (shippingCost +specialShipping) << endl;

cout << "Total of the order is: $" << ((temp\*COST\_PER\_SPOOL)+(shippingCost +specialShipping)) << endl;

}

int main()

{

const double COST\_PER\_SPOOL =100;

double shippingCost =10, specialShipping=0;

int numInStock = 10, numOrdered=0;

getOrderInfo(numOrdered,numInStock,specialShipping);

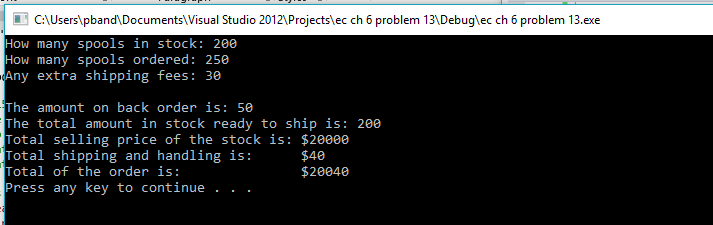
displayOrderInfo(numOrdered,numInStock,specialShipping,COST\_PER\_SPOOL,shippingCost);

system("pause");

return 0;

}

Output:



Program Challenge ch 7 problem 7

Code:

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

class Inventory

{

private:

int itemNumber;

int quantity;

double cost;

public:

Inventory()

{

itemNumber =0;

quantity =0;

cost= 0;;

}

Inventory(int item, int stock, double ask)

{

itemNumber= item;

quantity= stock;

cost = ask;

}

void setItemNumber(int x)

{

if(x > 0)

itemNumber =x;

}

void setQuantity(int i)

{

if(i > -1)

quantity =i;

}

void setCost(double c)

{

if(c >= 0)

cost =c;

}

int getItemNumber()

{

return itemNumber;

}

int getQuantity()

{

return quantity;

}

double getCost()

{

return cost;

}

double getTotalCost()

{

return quantity \* cost;

}

};

int main()

{

Inventory one;

Inventory two(10,10,5.00);

unsigned seed;

cout << "Program will display an item from user entered numbers" << endl<<endl;

double temp;

cout << "Please enter a positive number ";

cin >> temp;

one.setCost(temp);

cout << "Please enter a positve number ";

cin >> temp;

one.setItemNumber(temp);

cout << "Please enter a positive number ";

cin >> temp;

one.setQuantity(temp);

cout << "Results" <<endl << endl;

cout << "First Item" << endl;

cout << "Item number " << one.getItemNumber() << endl;

cout << "Quantity " << one.getQuantity()<< endl;

cout << "Cost $" << one.getCost() << endl;

cout << "Total $" << one.getTotalCost() << endl << endl;

cout << "Second Item" << endl;

cout << "Item number " << two.getItemNumber() << endl;

cout << "Quantity " << two.getQuantity() << endl;

cout << "Cost $" << two.getCost() << endl;

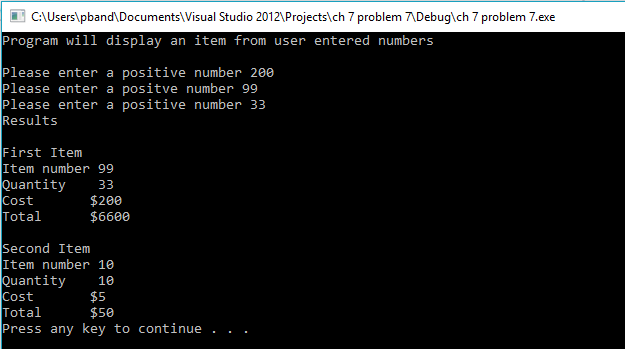
cout << "Total $" << two.getTotalCost() << endl;

system("pause");

return 0;

}

Output:



Program Challenge ch 8 problem 9

Code:

#include <iostream>

using namespace std;

class Stats

{

private:

double data[30];

int count;

public:

Stats()

{

count=0;

}

bool storeValue(double d)

{

bool open = false;

if(count < 30)

{

data[count] =d;

count++;

}

else

cout << "Out of Bounds" << endl;

if(count < 30)

open = true;

return open;

}

double total()

{

int total=0;

for(int x =0; x < count; x++)

{

total+=data[x];

}

return total;

}

double avg()

{

return total()/count;

}

double lowest()

{

double lowest = data[0];

for(int j=1; j<count; j++)

{

if(data[j] < lowest)

lowest = data[j];

}

return lowest;

}

double highest()

{

double highest = data[0];

for(int j=1; j<count; j++)

{

if(data[j] > highest)

highest = data[j];

}

return highest;

}

};

int main()

{

Stats rainfall;

double userEnter;

bool stop=true;

int x=1;

while(stop)

{

cout << "Enter the rainfall for the day of " << x << " : ";

cin >> userEnter;

stop = rainfall.storeValue(userEnter);

x++;

}

cout << "Rainfall Stats for one month" << endl<<endl;

cout << "Rainfall Total: " << rainfall.total() << endl;

cout << "Rainfall Average: " << rainfall.avg() << endl;

cout << "Rainfall lowest: " << rainfall.lowest() << endl;

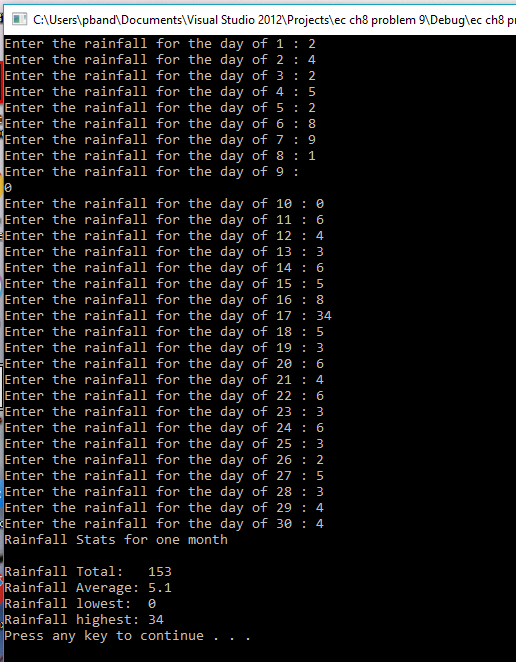
cout << "Rainfall highest: " << rainfall.highest() << endl;

system("pause");

return 0;

}

Output:



Program Challenge conversion Ch 8 problem

Code:

//ch8 problem 4 Monkey business

// animal weight

#include <iostream>

#include <iomanip>

#include <string>

using namespace std;

class Monkeys

{

private: string DAYS\_OF\_WEEK[7];

const int ROW =3;

const int COL =7;

double monkeys[ROW][COL];

public:

Monkeys()

{

DAYS\_OF\_WEEK ={"Sunday","Monday", "Tuesday","Wednesday","Thursday","Friday","Saturday"};

cout << "This program determines the amount of food the Moneys are eating" <<endl << endl;

}

void enterData()

{

for(int r =0; r < ROW; r++)

{

for(int c=0; c < COL; c++)

{

cout << "For monkey " << (r+1)<< " how much food was eaten on " << DAYS\_OF\_WEEK[c] << " :";

cin >> monkeys[r][c];

}

}

}

double getAvgEaten()

{

double avg =0;

int count =0;

for (int r=0; r < ROW;r++)

{

for(int c=0; c <COL; c++)

{

avg+=monkeys[r][c];

count++;

}

}

avg = avg/count;

return avg;

}

double getMostEaten()

{

double greatest;

greatest = monkeys[0][0];

for (int r=0; r < ROW;r++)

{

for(int c=0; c <COL; c++)

{

if(monkeys[r][c] > greatest)

greatest = monkeys[r][c];

}

}

return greatest;

}

double getLeastEaten()

{

double least;

least = monkeys[0][0];

for (int r=0; r < ROW;r++)

{

for(int c=0; c <COL; c++)

{

if(monkeys[r][c] < least)

least = monkeys[r][c];

}

}

return least;

}

};

int main()

{

Monkeys everywhere;

everywhere.enterData();

cout << endl << "The average amount eaten is " << everywhere.getAvgEaten << endl;

cout << "The greatest amount eaten is " << everywhere.getMostEaten << endl;

cout << "The least amount eaten is " << everywhere.getLeastEaten << endl;

system("pause");

return 0;

}

Output:

