**PLANNING/DESIGN DOCUMENTS:**

* VARIABLES (5 points)

**char** ch

**int** iNumShares

**float** fOrignalPricePerShare

**float** fCurrentPricePerShare

**float** fAmountSpent

**float** fAmountAfterSale

**float** fAmountLost

* FORMULAS (5 points)

fAmountSpent = iNumShares \* fOrignalPricePerShare;

fAmountAfterSale = iNumShares \* fCurrentPricePerShare

fAmountLost = (iNumShares \* fOrignalPricePerShare) – (iNumShares \* fCurrentPricePerShare)

* HIERARCHY (5 points)

Main

* IPO Chart - Input, Processing, Output (5 points)

|  |  |  |
| --- | --- | --- |
| **Input Variables** | **Process/Calculations** | **Output** |
| None | * fAmountSpent = iNumShares \* fOrignalPricePerShare * fAmountAfterSale = iNumShares \* fCurrentPricePerShare * fAmountLost = (iNumShares \* fOrignalPricePerShare) - (iNumShares \* fCurrentPricePerShare) | iNumShares  fAmountSpent  fAmountAfterSale  fAmountLost |

* **PSEUDOCODE** (10 points – minus 2 points for each syntax error and 5 points for each logic error)

main(){

Declare iNumShares as Integer = 750

Declare fOrignalPricePerShare as Float = 35.00

Declare fCurrentPricePerShare as Float = 31.15

Declare fAmountSpent as Float = iNumShares \* fOrignalPricePerShare

Declare fAmountAfterSale as Float = iNumShares \* fCurrentPricePerShare

Declare fAmoutLost as Float = (iNumShares \* fOrignalPricePerShare) - (iNumShares \* fCurrentPricePerShare)

Write “Kathryn bought “ + iNumShares + “ shares for a total of $” + fAmountSpent

Write "Kathryn then sold the shares for $" +fAmountAfterSale

Write "Kathryn lost a total of $" + fAmountLost

//end of program

}

* TEST DATA – 5 complete data sets (10 points)

Data is hard-coded.

750 \* 35 = 26,250

750 \* 31.15 = 23,362.50

26250 – 23,362.50 = 2887.50

* RAPTOR Flowchart (10 points – minus 2 points for each syntax error and 5 points for each logic error)
  + Pseudocode in lieu of Raptor per in-class discussion with Dr. Works on September 6th.

**SOURCE CODE (50 points)**

* Comments (10 points)
* Programming Style, proper usage of spaces and indents (15 points)
* Correctness (15 points – 5 points each error (syntax, logic))
* Output, overall quality and appearance (10 points)

//====================================================================

// Name : AdkinsDaron\_Project1.cpp

// Author : Daron Adkins

// Version : 1.0

// Description : Project 1 - Ch 2, Programming Challenge #15

//====================================================================

**#include** <iostream>

**#include** <iomanip> //to manipulate the numbers

**using** **namespace** std;

**int** **main**() {

//begin main module

**char** ch; //using this character to ensure program doesn't terminate until the user presses "Enter"

**int** iNumShares = 750;

**float** fOrignalPricePerShare = 35.00;

**float** fCurrentPricePerShare = 31.15;

**float** fAmountSpent = iNumShares \* fOrignalPricePerShare; //get total original amount spent

**float** fAmountAfterSale = iNumShares \* fCurrentPricePerShare; //net amount after sale

**float** fAmountLost = (iNumShares \* fOrignalPricePerShare) - (iNumShares \* fCurrentPricePerShare); //amount lost from the sale

//display calculations and ensure proper format for currency

cout << "\n";

cout << "Kathryn bought " << iNumShares << " shares for a total of $" << **fixed** << **setprecision**(2) << fAmountSpent << **endl**;

cout << "Kathryn then sold the shares for $" << **fixed** << **setprecision**(2) << fAmountAfterSale << **endl**;

cout << "Kathryn lost a total of $" << **fixed** << **setprecision**(2) << fAmountLost << **endl**;

cout << "\nKathryn needs to learn more about how stocks work.";

//hold until user presses Enter

cin.get(ch);

**return** 0;

//Fin

}

Note: Work submitted that does not adhere to the file naming standard outlined in class will NOT be graded…a 0 will be applied automatically.