CSCI140 – C++ language and Objects

Lab Assignments

Spring 2020

Instructions for your lab report.

1. You need to create a cover page formatted as following:
   (Notes: The cover page needs to be typed and printed)

CSCI140 – C++ language and Objects

MT SAC College

CSCI140

Lab #: ____________________

Description: ________________________

____

____

Due Date: _____________________

Name: _________________________

Grade: _________________________

Notes: ________________________

CSCI110 – Fundamentals of Computer Science

Requirements for your lab reports

Your submission should include:

1. The cover page
2. The listing of the source code (See details shown below)
3. A screenshot of your program execution

The source code should be organized and presented as:

1. Prolog
   a. Program Description
   b. Author
   c. Date
   d. Input variables
   e. Process Flow
   f. Output variables
2. A listing of source code with internal comments

Programming requirements:

1. Your program needs to be user-friendly and easy to understand.

2. You need to follow the book's and my instructions to code your program – no deviation. If you have any questions, please let me know.

CSCI140 – C++ Language and Objects

Lab 1A

Resource: C++ How to program

Requirements (non-OOP): Please work on p2.29 on page 71.

Test cases: Please produce a console message based on the book's instructions.

Due date: TBD

Submission includes 1) cover page 2) source listing 3) screen shot of program execution.

CSCI140 – C++ Language and Objects

Lab 1B

Resource: C++ How to program

Requirements (non-OOP): Please work on p2.31 on page 72.

Test cases: Here is your test case.

- Total Miles: 140
- Cost per gallon: $3.25
- MPG: 35
- Parking fee: $10
- Tolls fee: $10

Total transportation cost per day: ?

Please produce the console message based on the book’s instructions.

Due date: TBD

Submission includes 1) cover page 2) source listing 3) screen shot of program execution.

CSCI140 – C++ Language and Objects

Lab 1C

Resource: C++ How to program

Requirements (OOP): Please work on p3.9 on page 100.

Here is a list of test cases.

Test Cases:

1. Withdrawing $50 form Jane Green’s account. Printing new balance.
2. Withdrawing $150 form Jane Green’s account. Printing new balance.
3. Withdrawing $50 form John Blue’s account. Printing new balance.
4. Withdrawing $150 form John Blue’s account. Printing new balance.

Please produce the console message based on the book’s instructions.

Notes: If insufficient fund, print a message. Don’t deduct.

Due date: TBD

Submission includes 1) cover page 2) source listing 3) screen shot of program execution.

CSCI140 – C++ Language and Objects

Lab 2A

Resource: C++ How to program

Requirements (non-OOP): Please work on p4.19 on page 152. You need to use a while loop to implement this exercise and don’t assume anything. You can create your own list of 10 numbers and you need to promote the user for these numbers. Make sure that you echo your input values. Be user-friendly!

Test cases: Please produce a console message based on the book’s instructions.

Due date: TBD

Submission includes 1) cover page 2) source listing 3) screen shot of program execution.

CSCI140 – C++ Language and Objects

Lab 2B

Resource: C++ How to program

Requirements (OOP): Please work on p4.14 on page 150. You need to create a class “Account”. This class should contain the following methods and attributes. Be sure your project contain 3 files AccountTest.cpp, Account.cpp, and Account.h.

Methods:

- Constructor
- setBalance
- getBalance
- displayBalance
- setLimit
- getLimit
- displayLimit

Attributes

- account_number
- credit_limit
- balance

Test Cases:

- You need to create 3 accounts.

- Please produce the console message based on the book’s instructions.

Due date: TBD

Submission includes 1) cover page 2) source listing 3) screen shot of program execution.

CSCI140 – C++ Language and Objects

Lab 3A

Resource: C++ How to program

Requirements (Non-OOP): Please work on p6.34 on page 276.

Test Cases:

- Please follow the book's instructions to test your program.
- Please produce the console message based on the book's instructions.
- Please set the range of an interval to [1, 20] (not [1, 1000]) and set the limit of guessing to 3 times.
- You need to use RAND[] and SRAND[] to produce a different number and let the user play at least 3 times. When 2 out of 3 games the use guesses right, you need to post a well-done (i.e., congratulation!) message.

Due date: TBD

Submission includes 1) cover page 2) source listing 3) screen shot of program execution.

CSCI140 – C++ Language and Objects

Lab 3B

Resource: C++ How to program

Requirements (OOP): Please work on p7.21 on page 332. You need to create a class “Sales”. This class should contain the following methods and attributes. Be sure your project contains 3 files SalesTest.cpp, Sales.cpp, and Sales.h.

Methods:

- Constructor
- getTotalSales (Hint: Total of 8 sales by product by salesperson)
- displayTotalSale (Hint: See table below)
- anything else you need

Attributes

- Salesperson_name
- Product_number
- Total_sales

Test Cases:

- Create 3 salespersons and 3 products that are handled by each person.
- Create 8 sales per month for each month. You need to prompt the user for these 8 sales.
- Please produce a table as following:

April Sales Report (4/20/2020)										
Salesperson	Sales 1	Sales 2	Sales 3	Sales 4	Sales 5	Sales 6	Sales 7	Sales 8	Total
Product									
Johnny									
Product 1									
Product 2									
Product 3									
Robert									
Product 1									
Product 2									
Product 3									
Linda									
Product 1									
Product 2									
Product 3									

Due date: TBD

Submission includes 1) cover page 2) source listing 3) screen shot of program execution.

CSCI140 – C++ Language and Objects

Lab 4

Resource: C++ How to program

Requirements (Non-OOP): Please work on p8.12 on page 376. Here is a suggestion of Protypes.

```cpp
const int RACE_END = 70;

// prototypes
void moveTortoise(int* const);
void moveHare(int* const);
void printCurrentPositions(const int* const, const int* const);

default_random_engine engine{ static_cast<unsigned int>(time(0)) };
uniform_int_distribution<int> randomInt{ 1, 10 };
```

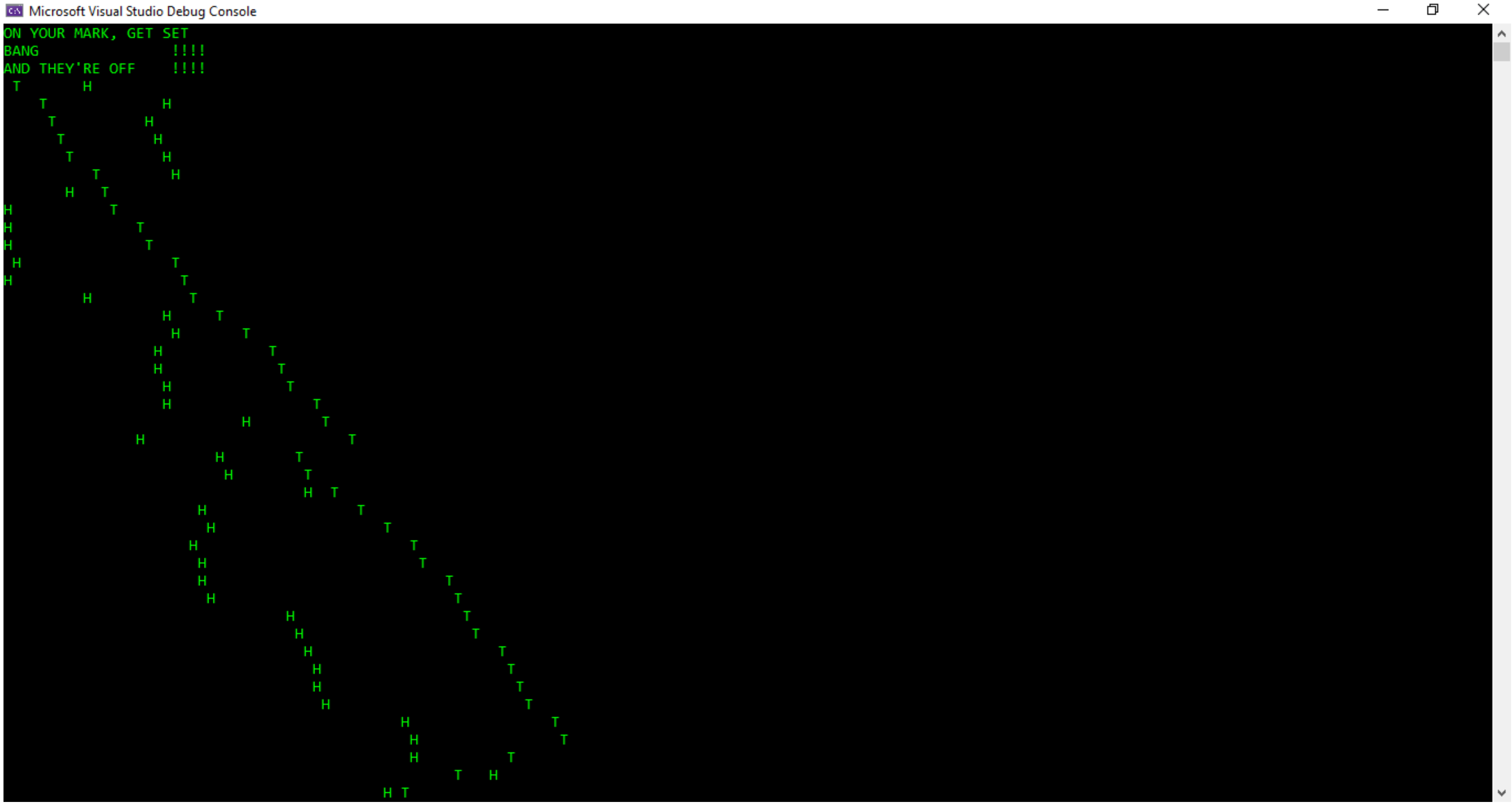
Test Cases:

- Please follow the book’s instructions to test your program.
- Please produce the console message based on the book’s instructions – see a simulation result shows on the next page as a reference.
- Please set the position numbers are in an interval of [1, 70].
- You need to use “random engine” and “uniform_int_distrribution” to produce a different move for Tortoise and Hare.
- Let the user play at least 2 times.

Due date: 5/22/2020

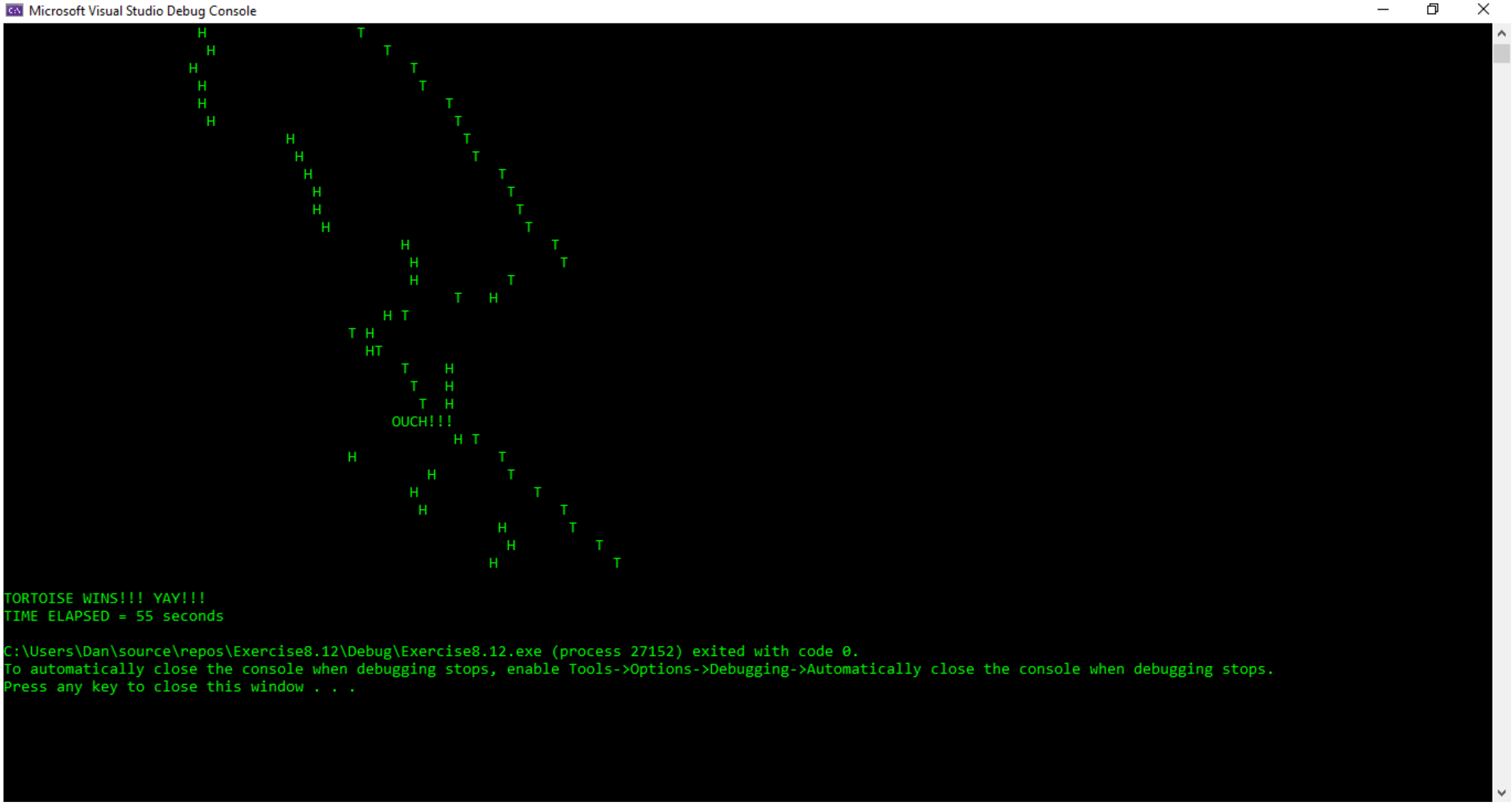
Submission includes 1) cover page 2) source listing 3) screen shot of program execution.

Result of the Simulation



```
ON YOUR MARK, GET SET
BANG               !!!!
AND THEY'RE OFF    !!!!
 T       H
    T             H
     T          H
      T          H
       T          H
          T        H
       H   T
H           T
H              T
H               T
 H                 T
H                   T
         H           T
                  H     T
                   H       T
                 H            T
                 H             T
                  H             T
                  H                T
                           H        T
               H                       T
                        H        T
                         H        T
                                  H  T
                      H                 T
                       H                   T
                     H                        T
                      H                        T
                      H                           T
                       H                           T
                                H                   T
                                 H                   T
                                  H                     T
                                   H                     T
                                   H                      T
                                    H                      T
                                             H                T
                                              H                T
                                              H          T
                                                   T   H
                                           H T
```

Continuation on the next page.



```
                      H                 T
                       H                   T
                     H                        T
                      H                        T
                      H                           T
                       H                           T
                                H                   T
                                 H                   T
                                  H                     T
                                   H                     T
                                   H                      T
                                    H                      T
                                             H                T
                                              H                T
                                              H          T
                                                   T   H
                                           H T
                                       T H
                                         HT
                                             T    H
                                              T   H
                                               T  H
                                            OUCH!!!
                                                   H T
                                       H                T
                                                H        T
                                              H             T
                                               H               T
                                                        H       T
                                                         H         T
                                                       H             T

TORTOISE WINS!!! YAY!!!
TIME ELAPSED = 55 seconds

C:\Users\Dan\source\repos\Exercise8.12\Debug\Exercise8.12.exe (process 27152) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```