

CSCI 201 – Computer Science 1

Outcomes. After completing this assignment, you should have:

1. Understood fundamentals of the problem-solving process.
2. Become comfortable with using Raptor to construct a flowchart.
3. Grasped the connection between **designing** a solution using a flowchart and **implementing** the designed solution using C++.
4. Learned how to document a program and report on the test results.

Homework Assignment.

Q 1. Creating a flowchart using Raptor.

Due date: Day 2 of Week 1

In pages 20 through 29 of the handout Introduction to the Problem Solving Process, we discussed the following problem: *Given an integer that represents the amount of change to be made, construct a flowchart that will print the number of quarters, dimes, nickels and pennies to be used, such that the total number of coins is minimized.*

This video explains the basic things about the Raptor program; Raptor is also discussed in the problem solving handout. The Raptor flowchart simulation program is available on all campus computers; can also be downloaded free from the internet at this link. Using Raptor, reconstruct the flowchart on Page 29 of the handout, and test your result. Place the raptor file in a folder **RaptorHW** and place the folder within the **Homework1** folder in your CourseFiles folder. The instructions for submitting by uploading to Coursefiles was explained in Lab 1.

Important: *All Mac users should run Raptor inside a Windows Virtual Machine. We cannot read/grade Mac files*

Q 2. Creating a C++ program for making change. **Due date: Day 3 of Week 1** Write a C++ program that makes change in US coins for a given amount. The program prompts the user to enter the amount for which we need change and then prints out the number of coins of each denomination needed to generate the amount, using the smallest possible number of coins. Here is a sample output:

```
Please enter the amount: 107
You will need:
4 Quarters
1 Nickel
2 Pennies
Thank-you, Goodbye!
```

Program Requirements. The code should be neatly indented, use meaningful variable names and named constants and have appropriate, succinct comments. Your program must not use any loop statements.

Grading scheme. This homework is worth 60 points. The specifications and points for each component are as follows:

1. **Raptor flowchart.** (10 points). Flowchart should run correctly.
2. **User Documentation.** (15 points) A neatly formatted document that addresses all of the following points in an **itemized** manner: (i) The purpose of the program (ii) Where the program is located, and how it is to be compiled and executed. (iii) Any particular issues that the user should be aware of. This includes any peculiarities of the program, and things the user should be careful about. The user document must be written such that a non-programmer can understand it.
3. **Script file showing the source, compilation and test runs.** (25 points) Using the script command start a script session, (see closed lab for details). Within this session, do the following:
 - Display the program (i.e., *source code*). This is done using the command: `cat <filename>`.
 - Compile the program.
 - Test the program. Testing is a critical part of software development (see the hand-out on Introduction to the Problem Solving Process, and section 1.6 of the text). Test cases are usually carefully designed to make sure that all aspects of the program are verified. You should carry out multiple tests to check various aspects of the behavior of the program.
 - Exit the script session.
4. **Test report.** (10 points) Report the test results in a testing table. See the Introduction to the Problem Solving Process for an example.
5. **Items to submit.** Create a folder **Homework1** in your CourseFiles folder. Place the Raptor flowchart file, the source code (the `.cpp` file), the script file, user document and test report in this folder.