

## Assignment #6

### Introduction to C Programming – COP 3223

#### **Objectives**

1. To learn how to use arrays to store and retrieve data to help solving problems.
2. Reinforce use of input files.

#### **Introduction: Pirate Time**

Your friend has not stopped talking about how cool pirates and how awesome it would be if they could be a real-life pirate captain! To amuse your friend, you have decided to create a series of programs about pirates.

#### **Problem: An Old Pirate's Treasure Chest (treasurechest.c)**

Your friend has recently come into possession of an old treasure chest. They are convinced there is great treasure inside. They may not even need to go to the Caribbean to get treasure!

The treasure chest is locked with seven locks and your friend has found 100 keys that might fit these locks. Your friend and their crew members will have to try different combinations of keys to try and get the treasure chest open.

There's one other thing. The treasure chest is haunted! The pirate who previously owned this treasure is still around, reminding other pirates that each lock has a unique key and letting them know if any of their current guesses are correct.

Your program will need to read in a file of the correct combination of keys, numbered 1-100. Then, prompt the user for the seven numbers indicating the seven keys they wish to use. They cannot use the same key twice in one attempt. If they have all the correct keys in the exactly correct order, they can open the chest. If they have some correct keys, regardless of order, let the user know how many keys are correct.

#### **Input File Format**

The input file will contain 7 unique integers from 0 to 100.

#### **Program Specification**

You must use arrays to solve the problem.

Your program should first prompt the user for the name of the input file. Then, your program should process the input file and copy the correct order of keys into the program. Then you can prompt the user for their first guess. If the user attempts to use a key more than once in a single guess, tell them they can only use each key once. If the user exactly matches the keys and the order, let them open the chest. If the user identifies some of the correct keys, let them know how many keys are correct but tell them they may not be in the right order.

#### **Output Sample**

Sample outputs will be provided on the webcourse.

**Deliverables**

One source file: *treasurechest.c* for your solution to the given problem submitted over WebCourses.

**Restrictions**

Although you may use other compilers, your program must compile and run using Code::Blocks. Your program should include a header comment with the following information: your name, course number, section number, assignment title, and date. Also, make sure you include comments throughout your code describing the major steps in solving the problem.

**Grading Details**

Your programs will be graded upon the following criteria:

- 1) Your correctness
- 2) Your programming style and use of white space. Even if you have a plan and your program works perfectly, if your programming style is poor or your use of white space is poor, you could get 10% or 15% deducted from your grade.
- 3) Compatibility – You must submit C source files that can be compiled and executed in a standard C Development Environment. If your program does not compile, you will get a sizable deduction from your grade.