CS161 Week3 Project Plan

Ziwei Wu

Student Number: 933296824

July 9, 2017

3A. Write a program that asks the user how many integers they would like to enter. You can assume they will enter a number >= 1. The program will then prompt the user to enter that many integers. After all the numbers have been entered, the program should display the largest and smallest of those numbers.

Testing Cases

Input	Output
1, 2, 3, 4	min:1, max:4
999, 1, -1223231231, 999, -122323123	min: -122323123, max: 999
5668, 5668, 5668	min: 5668, max: 5668
-1001, -1, 0, - 506	min: -1001, max: 0
0	min: 0, max: 0

Pseudocode:
Ask the user how many integers he/she wants to enter.
Initialize an variable called $InputNum$ to store this value.

Initialize two variables called $minNum$ and $maxNum$.
Initialize a variable called num.

Ask the user to enter $InputNum$ number of integers.
Store the first input to num.
Set both $minNum$ and $maxNum$ to the value of num.

for from zero to $InputNum - 1$ **do**
  Reinitialize num to zero
  Store the input to num
  if $num < minNum$ **then**
    Set $minNum$ to num
  else if $num > maxNum$ **then**
    Set $maxNum$ to num
  end if
end for

Display the variable $minNum$ and $maxNum$ to the user.

3B. Write a program that prompts the user for the name of a file and then tries to open it. If the input file is there and can be opened, the program should read the list of integers in the file, which will have one integer per line. The program will then add together all the integers in the file, create an output file called sum.txt, and write the sum to that file (just that number - no additional text).

Testing Cases

Input	Output
file with list of integers: "number.txt"	Sum of integers is written to file: "sum.txt"
file cannot be open	"file can not be opened"
file is not there	"file is not there"
file does not contain integers	"file does not contain integers"

Pseudocode:
Ask the user to enter a file name.
Set a variable called *sum* and initialize it to zero.
Read the file.

if file can not be opened **then**
  Print a message to user and terminate the program
else if file is not there **then**
  Print a message to user and terminate the program
else if file doesn't contain list of integers **then**
  Print a message to user and terminate the program
else
  for Each line in the file until the end of file **do**
    read the integer and add it to *sum*
  end for
end if

Output the value of *sum* to file named "sum.txt"

3C. Write a program that prompts the user for an integer that the player (maybe the user, maybe someone else) will try to guess. If the player's guess is higher than the target number, the program should display "too high" If the user's guess is lower than the target number, the program should display "too low" The program should use a loop that repeats until the user correctly guesses the number. Then the program should print how many guesses it took.

Testing Cases

Input	Output
1	Too Low - try again
10000	Too High - try again
5000	Too High - try again
2500	You guessed it right in 2 tries

Pseudocode:
Initialize a variable called *answer* and set it to zero.

Ask the user to enter a number as the answer.
Assign the input number to variable *answer*.

Initialize a variable called *guess* and set it to zero to store the guessed number.
Initialize a variable called *count* and set it to zero to store the number of times that player has guessed.

Ask the player to enter a number to make a guess and set *guess* to that input value.
Increment the *count* by 1.

while *guess* does not equal to *answer* **do**
  if *guess* < *answer* **then**
    Tell the player the guess is too small
    Ask for another guess and assign to *guess* variable
    Increment the *count* by 1
    Jump back to the beginning of while loop
  else if *guess* > *answer* **then**
    Tell the player the guess is too large
    Ask for another guess and assign to *guess* variable
    Increment the *count* by 1
    Jump back to the beginning of while loop
  end if
end while

Tells the player that he/she has made the right guessed.
Print the *count* to tell the player how many times he/she guessed.