Samuel Park

Tuesday Loops

1. This is an infinite loop count will always be greater than 0 both because it is not being decremented (it is being incremented) and the initial value is greater than 0. The output is 10, 11, 12, and so on.
2. This is a normal for-loop that will parse through the array until the value of the index ‘I’ equals the string “Waldo”. In this case, the index “wheresWaldo” is 3. The output is 3.
3. This is an enhanced for-loop or for-each loop that parses through each char of the char array to check for the value ‘I’. Since there is no I in “team”, it would keep the Boolean value to false. The output is false.
4. This is a for-loop with the index initialized to 0 incrementing until it reaches 9. The if conditional will check if the index is even, print the index, and continue to check for each index. The output is 2, 4, 6, 8.

/\*

This is after checking the code, but I realized that continue doesn’t mean the program will continue with printing the evens, it means it stops running through the code to continue with the loop. So, the output is 1, 3, 5, 7, 9.

\*/

1. This is a for-loop with the index initialized to 0 incrementing by 2 until reaching 9. At each increment, the loop will print the index. The output is 0, 2, 4, 6, 8.
2. This is a for-loop with the index initialized to 100 decrementing until it is less than 0. It will print the numbers from 100 to 1 in descending order.

/\*

This is after checking the code, but I realized that the condition for the index being less than 0 is never met from the start, so the loop will not run at all. No output.

\*/

1. This is an enhanced for-loop or for-each loop that parses through each string of the array birds. It has a conditional statement checking each string to see if it is equal to the string “goose”. If it is a goose, the program interjects a break to return to parsing the array. Otherwise, if the string is “duck”, it increments the count of ducks. The output is “There are 5 ducks.

/\*

This is after checking the code, but I realized that the keyword break even when it is within the conditional, will break the loop from running too. So, there are only 3 ducks before reaching the goose, but if the keyword is replaced with continue, then the loop would continue even after running into the goose.

\*/