

# CS 200 Sections 02 & 04 Spring 2013

## Week #11: Top 3 Lessons Learned

Three things that I learned are that when you are code tracing you need to pay attention to the current values of the variables in a program. The values of the variables can change as the program progresses. It is a good idea to make a list with the value of each variable and make changes as you go along.

Second, when you are code tracing, you need to pay attention when you are dealing with a nested loop. It is a good idea to mark where the inner loop starts and where it ends. This will also help you to determine when you exit that inner loop and continue with the outer loop.

Third, on the final, when you are given an output sample for a coding problem, make sure that the output prompts you produce with your code is the same as the sample that you were given. Also, remember...DO NOT HARD CODE SPECIFIC SOLUTIONS! Your code should work for

J. Gomez

1. A portion of a program that repeats a statement or group of statements is called a loop. The statement or group of statements to be repeated in a loop is called the body of the loop. Each repetition of the loop body is called an iteration of the loop.

2. -----  
while (Boolean\_Expression)  
{  
  First\_Statement  
  Second\_Statement  
  ...  
  Last\_Statement  
}

-----  
The Semantics of the do-while Statement  
The do-while Statement SYNTAX

```
do  
{  
  First_Statement  
  Second_Statement  
  ...  
  Last_Statement  
} while (Boolean_Expression);
```

-----  
3. A loop that iterates its body repeatedly without ever ending is called an infinite loop.

D. Mirdadi

1.) One way to end up getting into an infinite loop error is by forgetting MOE. Make sure you collect a new value for the variable used in the conditional statement that can potentially cause the conditional statement to be false so the loop terminates.

2.) Remember that when you repeat an INNER FOR loop you always start at the top! That means that all variables are reset to the value the loop originally had set them to.

3.) Do not be afraid to use variables if you feel you need to use them. Just as there are more than one way to catch a fish, there is more than one way to program and get the same results. As long as you get the problem solved, the method doesn't matter. (for now)

## E. Herring

1. For loop repeats until the conditional statement is false:

```
for(int i = 0; i < 5; i++) {  
    //perform functions within the loop, i.e. other for loop;  
}
```

That's why we can print multiple characters in one line.

2. To better understand nested for loops I would recommend spending some time on drawing patterns (hope the patterns will not be on the final - fingers crossed) like this one:

```
public class PatternsForLoopsLessonLearned
```

```
{  
    public static void main(String[] args)  
    {  
        for(int a = 10; a > 0; a--)  
        {  
            System.out.print("|");  
            for(int b = 10; b > a; b-- )  
            {  
                System.out.print("|");  
            }  
            System.out.print("_");  
  
        }  
        System.out.print("*");
```

```
        for(int c = 1; c < a; c++ )  
        {  
            System.out.print("_");  
            System.out.print("|");  
  
        }  
        System.out.print("|");  
        System.out.println();  
    }  
}
```

```
/*-----OUTPUT-----
```

```
----jGRASP exec: java PatternsForLoops
```

```
|*_|_|_|_|_|_|_|_|_|_|_|_|_|_|
|_|_*_|_|_|_|_|_|_|_|_|_|_|_|_|
|_|_|_*_|_|_|_|_|_|_|_|_|_|_|_|
|_|_|_|_*_|_|_|_|_|_|_|_|_|_|_|
|_|_|_|_|_*_|_|_|_|_|_|_|_|_|_|
|_|_|_|_|_|_*_|_|_|_|_|_|_|_|_|
|_|_|_|_|_|_|_*_|_|_|_|_|_|_|_|
|_|_|_|_|_|_|_|_*_|_|_|_|_|_|_|
|_|_|_|_|_|_|_|_|_*_|_|_|_|_|_|
|_|_|_|_|_|_|_|_|_|_*_|_|_|_|_|
|_|_|_|_|_|_|_|_|_|_|_*_|_|_|_|
```

```
----jGRASP: operation complete.
```

```
-----*/
```

3. Hmm did I learn anything else this week? Tracing! Keep track of every single step. Remember about semicolon. Example from the book:

Ex. 9. What output is produced by the following code?

```
for (int n = 4; n > 0; n--)
    System.out.println(n);
```

Ex. 10. What output is produced by the following code?

```
for (int n = 4; n > 0; n--); <==== SEMICOLON!
    System.out.println(n);
```

(This is not the same as the previous question. Look carefully.)

M. Mardosz