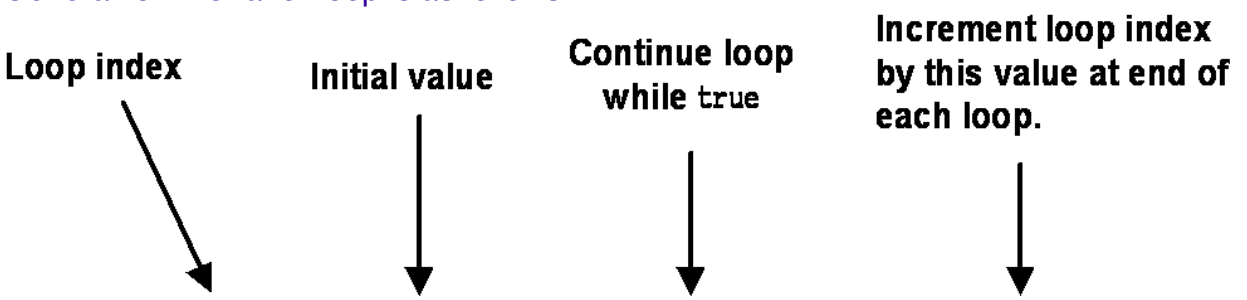
CSE20 : Lab #10 - Loops

Overview

This week we are going to learn to use different Loop constructs to write dynamic programs that change behavior based on the type of inputs they receive from the user. We will start with exercises on how to convert from while-loop to for-loop and do-while-loop. Then try to find the appropriate loop type to solve different problems.

For-loop Syntax

General form for a for-loop is as follows:




```
for ( index = initExpr; continueExpr; incrementExpr ) {
   Statement 1;
   ...
   Statement n;
}
```

The sequence of execution follows these set of rules:

- Before starting:
  1. *initExpr* is set

- if *continueExpr* is true
  1. Run the body statements.
  2. Execute the *incrementExpr*.
  3. Go to evaluating *continueExpr*

- Loop ends when *continueExpr* is false

We can put any number of statements inside the body of the loop. If it's only one statement then we don't need block {} curly brackets, otherwise we need them just like if statements. Be sure to understand each of the three parts in a for-loop statement.

One statement form for a for-loop is as follows:

```
for ( initExpr; continueExpr; incrementExpr)
        statement;
```

Do-While-loop Syntax

One statement form for a Do-while-loop is as follows:

```
do
        statement;
while (booleanExpr);
```

General form for a while-loop has a block statement as the body:

```
do {
        statement 1;
        statement 2;
        …
        statement n;
} while (booleanExpr);
```

(Reading) Chapter 4.5, 4.8 & 4.9

- Answer Participation Activity 4.5.2, 4.5.3, 4.5.4 & 4.5.5
- Answer Participation Activity 4.8.1 & 4.8.2
- Answer Participation Activity 4.9.2

Getting started

After starting Eclipse, create a new project called Lab 10. Refer to lab9's AllNumWhile.java for how to write while-loop. Download AllNumFor.java and AllNumDo.java from the assignment page for examples on how the same program can be written in 3 different ways.

(Exercise) Convert – SumAllFor.java & SumAllDo.java

Use your lab9 program SumAll that is written as While-loop as reference. Then create two new files and a different loop type in each. SumAllFor.java should contain a for-loop to implement the same logic. SumAllDo.java should behave similarly using a do-while-loop. Refer to your reading assignment and the examples to note the differences and convert correctly.

(Exercise) Create – Multiples.java (for-loop)

This program should ask the user for the maximum number to print out to and then print each ***multiple*** of the base number starting from the base to the maximum. For-loop is the most appropriate for this type of program since it has an obvious starting value and ending value.

```
Please enter the max number:10
Enter the Base: 2
Number is 2
Number is 4
Number is 6
Number is 8
Number is 10
```

```
Please enter the max number:18
Enter the Base: 3
Number is 3
Number is 6
Number is 9
Number is 12
Number is 15
Number is 18
```

(Exercise) Create – SumSquare.java (any loop)

This program should ask the user for the maximum number to print out to and then print each number starting from 1 to the maximum along with it squared. It will also print out the sum of all the square numbers like this:

```
Please enter the max number:3
Number 1 squared is 1
Number 2 squared is 4
Number 3 squared is 9
Sum of squares is 14
```

```
Please enter the max number:10
Number 1 squared is 1
Number 2 squared is 4
Number 3 squared is 9
Number 4 squared is 16
Number 5 squared is 25
Number 6 squared is 36
Number 7 squared is 49
Number 8 squared is 64
Number 9 squared is 81
Number 10 squared is 100

Sum of squares is 385
```

(Exercise) Create PosAverage.java (do-while-loop)

Implement the following using a do-while-loop since it will ask for a number at least one time. That is a prototypical reason for using a do-while-loop over the other two type of loops. Program asks the user to enter any number then do the following:

- It adds it to the total if the number is positive
- Then repeat to ask for another number
- If the number is not positive then it stops
- Print the average of all the positive numbers

Sample Output:

```
Enter 0 number: 10
Enter 1 number: 3
Enter 2 number: 4
Enter 3 number: 11
Enter 4 number: -50
Average is 7
```

```
Enter 0 number: 2
Enter 1 number: 0
Average is 2
```

```
Enter 0 number: 324
Enter 1 number: 23
Enter 2 number: 32
Enter 3 number: 51
Enter 4 number: 54
Enter 5 number: 42
Enter 6 number: 95
Enter 7 number: 85
Enter 8 number: 84
Enter 9 number: 325
Enter 10 number: -2
Average is 111
```

(Assessment) Logic Check

1) How many iterations for each loop where max = 10 and incr = 3
   a) for (int i = 0; i < max; i++)
   b) for (int i = 0; i < max; i += 2)
   c) for (int i = 0; i < max; i += incr)
   d) for (int i = incr; i < max; i += incr)
   e) for (int i = max; i > 0; i--)

f) for (int i = max; i < 0; i--)

g) for (int i = 1; i < max; i *= incr)

h) for (int i = 0; i < max; i *= incr)

2) How will *while(true)* behave if the loops continue as long as the condition is true?

3) When should you use

a) For-loop?

b) Do-while loop?

What to hand in

When you are done with this lab assignment, you are ready to submit your work. Make sure you have done the following ***before*** you press Submit:

- Answers to Participation Activity 4.5.2, 4.5.3, 4.5.4, 4.5.5, 4.8.1, 4.8.2 & 4.9.2
- Include answers to Assessment questions
- Attach converted SumAllFor.java and SumeAllDo.java
- Attach created Multiples.java, SumSquare.java and PosAverage.java
- List of Collaborators