

CISC124 – Today's Topics

- More on while loops
- for loops
- switch statement
- Formatted console output

Winter 2019 F. de la Parra 1

•1

while Loop Exit

```
while (Expression) {
    Block 1 Statements
    Block 2 Statements
}
```

Loop terminates when (Expression) evaluates to false

```
while (Expr1) {
    Block 1 Statements
    if (Expr2) break;
    Block 2 Statements
}
```

Loop terminates when (Expr1) or (Expr2) evaluates to false

- If Expr1 is always true. Example → (true)
 - Generally, not good style. No explicit loop termination condition.
 - If Expr2 is always true → Infinite loop!
- Regardless of Expr1's value, if Expr2 is always true → dead code (code that is never executed! → Block 2 Statements)

Winter 2019 F. de la Parra 2

•2

for Loop

```
for (Init Expr; Comp Expr; Updt Expr) {
    Block of Statements
}
```

```
for (Init Expr; Comp Expr; Updt Expr)
    Single Statement;
```

Loop terminates when Comp Expr evaluates to false

- Init Expr → declare and initialize variables used in loop. Example: int i=0, j= p.length()
- Comp Expr → check expression on loop variables. Example: (i<5 && i<j)
- Updt Expr → check expression on loop variables. Example: i++, j--

Winter 2019 F. de la Parra 3

•3

for Loop Example

```
String p="hello world";
for (int i=0,j=p.length(); (i<5 && j >= i); i++,
    p=p.length()) {
    System.out.println("Current phrase is: " + p(i,j));
}
```

Current phrase is: hello world

Current phrase is: ello worl

Current phrase is: llo wor

Current phrase is: lo wo

Current phrase is: o w

Winter 2019 F. de la Parra 4

•4

for Loop Nesting

```
for (int i=0; i<5; i++) {
    for (int j=i; j<5; j++) {
        System.out.print(j + " ");
    }
    System.out.println();
}
```

- for loops can be nested to any level
- Variables defined in an outer for loop are visible in inner for loops

Output →

```
0 1 2 3 4
1 2 3 4
2 3 4
3 4
4
```

Winter 2019 F. de la Parra 5

•5

for each loop

```
int[] nums = {1,2,3,4,5,6,7,8,9};
int sum = 0;
for (int item : nums)
    sum += item;
System.out.println("Sum is " + sum);
```

Output → 45

- "for each" loop iterates over each and all the elements of a collection (example array of primitive or class types)
- Iterator variable ("item", in example), of the same type as the collection base type ("int" in example), takes the value of a value in the collection, at each pass of the loop.

Winter 2019 F. de la Parra 6

•6

for Loop Exit

- Similar to while loop
 - Loop terminates when CompExp is false
- "break" terminates current loop and exits to previous nesting loop (if one exists)
- "continue" returns execution to the beginning of the loop
- Unconditional "break" or "continue" will produce dead code (if it's not the last statement in loop's body)
- for (; ;) {body} is an infinite loop if no conditional break is executed in the loop's body

Winter 2019

F. de la Parra

7

•7

switch statement

```
switch (Expression) {
    case value1 : {
        Block 1 Statements
        break;
    }
    ...
    case valueN : {
        Block N Statements
        break;
    }
    default : {
        Default Block of Statements
        break;
    }
}
```

- Expression evaluates to values of the following primitive types: byte, short, char, int
- In can also evaluate to wrapper class types: Byte, Short, Character, Int
- Statements in "case block" whose value matches Expression's value are executed
- If no value matches Expression's value, then the statements in the "default" block are executed.
- If no break statement is found, execution continues with next "case block" until the end of switch statement, or until next break is found
- switch also works with "enum" types

Winter 2019

F. de la Parra

8

•8

Formatted Console Output

System.out.printf("format-string",[arg1, arg2...]);

format-string = [string1]FS1[string2]FS2[string3]...

Format Specifier:

FS = % [flags] [width] [.precision] conversion-character

flags:

- : left-justify
- + : number sign
- 0 : zero padding
- , : digit group separator
- : space will display sign

conversion-character

- d : integer (byte,short,int,long)
- f : number sign
- c : character
- s : string
- h : hashcode
- n : newline

Winter 2019

F. de la Parra

9

•9

F. de la Parra

2