Nested Loops

Introduction to Programming

Nested Control Structures

- Any control structure can be nested within another
- Nested control structures can involve both iterative and selection statements
- ***i.e.*** a loop can be nested within an `if else` and an `if else` can be nested within a loop
- The nested structures may be as complex as is necessary

`if` nested in a `for`

```
int total = 0;
for(int i = 1; i <= 100; i++)
{
   if(i % 2 == 0)
   {
        total = total + i;
   }
}
System.out.print("Total: " + total);
```

```
/*use if and do-while to ensure user enters
  correct exam result */

do{
  System.out.print("Enter an exam result: " );
  result = keyIn.nextInt();

  if(result < 0 || result > 100)
  {
     System.out.print("Invalid result");
  }

}while(result < 0 || result > 100);
```

Nested loops

- Loops like if-else statements can be nested, one within another.
- The *inner* and *outer* loops need not be generated by the same type of control structure.
- It is essential that one loop be completely embedded within the other – there can be no overlap
- Each loop must be controlled by a different counter variable.

Nested `for` loops

- To print one star on the screen we can use a statement as follows

```
System.out.print("*");
```

- To print 5 stars on screen

REPEAT 5 times

DISPLAY one star

- Repeat `System.out.print("*");` 5 times.

`for` loop to print 5 stars

```
for(int i=1; i=<5; i++)
{
    System.out.print("*");
}
```

Output

*****

Nested `for` loops

- If we want to display the following pattern

- We want to print

*****

5 times

* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

REPEAT 5 times

REPEAT 5 times

DISPLAY 1 star

```
for(int i=1; i<=5; i++)
{
  System.out.print("*");
}
```

Repeat this 5 times – so we put it inside another loop

```
for(int outer=1; outer<=5; outer++)
{
  for(int inner=1; inner<=5; inner++)
  {
      System.out.print("*");
  }
  System.out.println();
}
```

inner

outer

```
for(int inner=1; inner<=5; inner++)
{
    System.out.print("*");
}
```

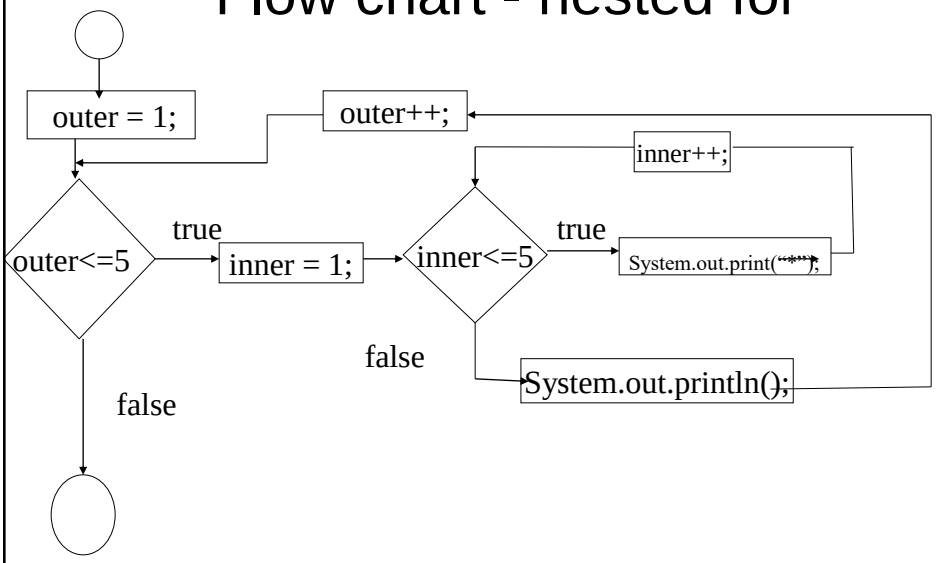
inner

```
for(int outer=1; outer<=5; outer++)
{
  for(int inner=1; inner<=5; inner++)
  {
      System.out.print("*");
  }

  System.out.println();
}
```

outer

Flow chart - nested for



- Consider printing the numbers 1 to 5

  1 2 3 4 5

- REPEAT 5 times

  DISPLAY current value

```
for(int j = 1; j <= 5; j++)
{
   System.out.print(j + " ");
}
```

- To print

  1 2 3 4 5
  1 2 3 4 5
  1 2 3 4 5

REPEAT 3 times
  REPEAT 5 times
    DISPLAY current value

```
for(int j = 1; j < 6; j++)
{
        System.out.print(j + " ");
}
```

- Use the following code

```
for(int i = 1; i <= 3; i++)
{
    for(int j = 1; j <= 5; j++)
    {
          System.out.print(j +" ");
    }
    System.out.println();
}
```

inner loop repeats 5 x 3 times

outer loop iterates 3 times

- Consider the following pattern

  11111
  22222
  33333
  44444
  55555

- Use the following code

```
for(int i = 1; i <= 5; i++)
{
    for(int j = 1; j <= 5; j++)
    {
        System.out.print(i +" ");
    }
    System.out.println();
}
```

Example Multiplication tables

- To display 12 multiplication tables – each consisting of 12 multiplications

```
1*1 = 1
1*2 = 2
  :
1*12  = 12
2* 1 = 2
2* 2 = 4
  :
2*12 = 24
  :
12 *1 = 12
12* 2 = 24
  :
12*12 = 144
```

To display 1 set of tables:

```
for(int j = 1; j <= 12; j++)
{
   System.out.print(i +"*" +j +"=" +(i*j) +" ");
}
```

To display 12 by 12 set of tables:

```
for(int i = 1; i <= 12; i++)
{
   for(int j = 1; j <= 12; j++)
   {
       System.out.print(i + " * " + j + " = " + (i*j) + " ");
   }
   System.out.println();
}
```