



NESTED LOOPS

Deborah A. Trytten
CS 1323/1324

NESTED: LOOP WITHIN A LOOP

Critical rule

- Within each single trip through the outer loop the inner loop must repeat in its entirety

One way to separate the outer loop from the inner loop is to consider which loop is most active

- Outer loop plods along
- Inner loop moves like crazy

EXAMPLE 1: INDEPENDENT LOOPS

```
int size = 4;
int sum = 0;
while (size > 0)
{
    for (int j = 0; j < 3; ++j)
    {
        sum = sum + j;
    }
    size = size - 1;
}
// Rectangular shape
```

EXAMPLE 2: DEPENDENT LOOPS

```
int[] data = {1, 3, 7, 5, 9}
for (int outer = 0; outer < data.length; ++outer)
{
    for (int inner = outer+1; inner < data.length; ++inner)
    {
        if (data[outer] > data[inner])
        {
            return false;
        }
    }
}
return true;
// Triangular shape
```

THINK PAIR SHARE

Trace the execution of the following code:

```
// Note: oversize array (capacity is 4; size is 3)
String [] names = {"Abby", "Raven", "Jazz", "Daisy"};
int nameSize = 3;

for (int idx1 = nameSize-1; idx1 >= 0; --idx1)
{
    for (int idx2 = nameSize-2; idx2 >= 0; --idx2)
    {
        names[idx1] = names[idx2];
    }
}
```

IClicker Question

What is in sum when the following loop is finished executing?
Drawing a table may help!

```
int sum = 0;
for (int i = 0; i < 4; ++i)
{
    for (int j = 0; j < 3; ++j)
    {
        sum = sum + i*j;
    }
}
```

- a) 6 b) 12 c) 18 d) 24 e) Something else

RECOGNIZE NESTED LOOPS

Single loops are often linear

- Linear search, marching through an array
- Binary search is an exception

Nested loops are often rectangular or triangular

- Rectangular shapes have independent loops
- Triangular shapes have the inner loop dependent on the outer loop

Key to implementation success: know what the inner loop does

- Paraphrase its purpose
- Wrap the outer loop around it

EXAMPLE: READ GRADE SPREADSHEET

Suppose we have a file that contains some grade data.

Format:

- #rows #columns<newline>
- grade11 grade12 grade13 grade14 Name1<newline>
- grade21 grade22 grade23 grade24 Name2<newline>
- ...

Find the average of all the grades using a nested loop.

Details:

- What does the inner loop do? → Reads all the grades on a single row.
- Rectangular or triangular? → The inner loop executes the same number of times for each outer loop execution.

EXAMPLE: INTERSECTION OF TWO SETS

Given two arrays of unique integers, write a method that returns the integers that appear in both arrays.

“Unique integers” means that no integer appears twice in the same array.

- Each array is like a set (in mathematics).
- This method is analogous to finding the intersection of two sets.

Details:

- Signature? → Perfect size arrays are implied.
- What does the inner loop do? → An integer in the first array is compared to all the integers in the second array.
- Rectangular or triangular? → The inner loops executes the same number of times for each outer loop execution (assuming we don't break early).

IClicker QUESTION

Suppose we want to write a method that takes two arrays of unique String objects and returns the Strings that appear in both arrays. (Assume we don't break early if a String is found in both arrays.)

```
String[] intersection(String[] first, String[] second)
```

- a) This should be done with independent nested loops.
- b) This should be done with a single loop.
- c) This should be done with dependent nested loops.
- d) This doesn't require a loop.

IClicker Question

Suppose we want to write a method that determines whether an array of String objects is unique (i.e., if the array has any duplicate elements). Assume the elements are unsorted and that the method does not sort them.

```
boolean hasDuplicates(String[] array)
```

- a) This should be done with a single loop.
- b) This should be done with a rectangular nested loop.
- c) This should be done with a triangular nested loop.
- d) No loop is necessary.