

## L6. More on Iteration

Using a Count Variable

Developing For-Loop Solutions

Nested Loops

## A Year-Printing Fragment

```
First = input('Enter first year');  
Last = input('Enter last year');  
for y = First:Last  
    fprintf('%5d\n',y)  
end
```

### How It Works

```
for y = First:Last  
    fprintf('%5d\n',y)  
end
```

1999	2001	
First	Last	y

Suppose First is 1999 and Last is 2001.

### How It Works

```
for y = First:Last  
    fprintf('%5d\n',y)  
end
```

1999	2001	1999
First	Last	y

We enter the for-loop and y is initialized

### How It Works

```
for y = First:Last  
    fprintf('%5d\n',y)  
end
```

1999	2001	1999
First	Last	y

Is the value in y  $\leq$  than the value in Last?

### How It Works

```
for y = First:Last  
    fprintf('%5d\n',y)  
end
```

1999	2001	2000
First	Last	y

Yes. Execute the loop body and increment y.

Output  
1999

### How It Works

```
for y = First:Last
    fprintf('%5d\n',y)
end
```

Output  
1999

1999	2001	2000
First	Last	y

Is the value in y <= the value in Last?

### How It Works

```
for y = First:Last
    fprintf('%5d\n',y)
end
```

Output  
1999  
2000

1999	2001	2001
First	Last	y

Yes. Execute the loop body and increment y.

### How It Works

```
for y = First:Last
    fprintf('%5d\n',y)
end
```

Output  
1999  
2000

1999	2001	2001
First	Last	y

Is the value in y <= the value in Last?

### How It Works

```
for y = First:Last
    fprintf('%5d\n',y)
end
```

Output  
1999  
2000  
2001

1999	2001	2002
First	Last	y

Yes. Execute the loop body and increment y.

### How It Works

```
for y = First:Last
    fprintf('%5d\n',y)
end
```

Output  
1999  
2000  
2001

1999	2001	2002
First	Last	y

Is the value in y <= the value in Last?

### How It Works

```
for y = First:Last
    fprintf('%5d\n',y)
end
```

Output  
1999  
2000  
2001

1999	2001	2002
First	Last	y

No. The loop is finished.

## Problem Solving With the For-Loop

```
for count variable = expression for starting value : expression for ending value  
    The calculation to be repeated.  
end
```

## Developing For-Loop Solutions

Illustrate the thinking associated with the design of for-loops

The methodology of stepwise refinement.

An example..

## A Game: TriStick

Pick three sticks each having a random length between zero and one.

You win if you can form a triangle whose sides are the sticks. Otherwise you lose.

Win:



Lose:



## Problem

Estimate the probability of winning a game of TriStick by simulating a million games and counting the number of wins.

## Pseudocode

Initialize running sum variable.

Repeat 1,000,000 times:

    Play a game of TriStick by picking the three sticks.

    If you win

        increment the running sum

Estimate the probability of winning

## Refine...

```
% Initialize running sum variable.  
wins = 0;  
for n = 1:1000000  
    Play the nth game of TriStick by  
        picking the three sticks.  
    If you win  
        increment the running sum.  
end  
% Estimate the prob of winning  
p = wins/1000000
```

## Refine the Loop Body

```
Play the nth game of TriStick by  
picking the three sticks.  
If you win  
    increment the running sum.
```



## Refine the Loop Body

```
% Play the nth game of TriStick  
% by picking the three sticks.  
a = rand; b = rand; c = rand;  
  
if (a<b+c) && (b<a+c) && (c<a+b)  
    % No stick is longer than the  
    % sum of the other two.  
    wins = wins+1;  
end
```

## Key Problem-Solving Strategy

Progress from pseudocode to Matlab through a sequence of refinements.

Comments have an essential role during the transitions. They "stay on" all the way to the finished fragment.