
CS1371

Introduction to Computing for Engineers

Iteration

1

9/4/2003

Control Flow Statements

Learning Objectives

Learn about how to control the sequence of expressions that are evaluated in a program.

Topics

- **FOR** statements
- **WHILE** statements

2

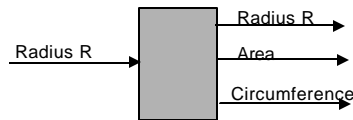
Problem 1: Introducing the FOR Structure

Step 1: Describe problem: Write a MatLab program to

- calculate the area and the circumference of **TEN** circles
- enter the radius as an input variable
- output radius, area and circumference **IF** the area is greater than 20 square units.
- output the number of circles with area ≥ 20 .

Step 2: Describe input and output

INPUT CALCULATE OUTPUT



Step 3: Define test cases

- $R=0 \rightarrow \text{Area}=0$ and $\text{Circumference}=0$
- $R=1 \rightarrow \text{Area}=\pi$ and $\text{Circumference}=2\pi$

3

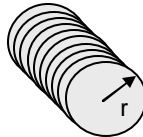
Problem 1 ... cont'd.

• Step 4: Develop the solution

– Describe the algorithm:

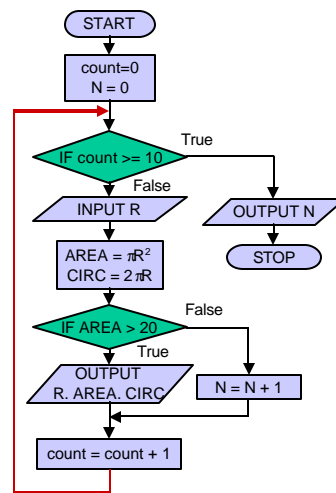
$$\text{Area} = \pi r^2$$

$$\text{Circum} = 2\pi r$$



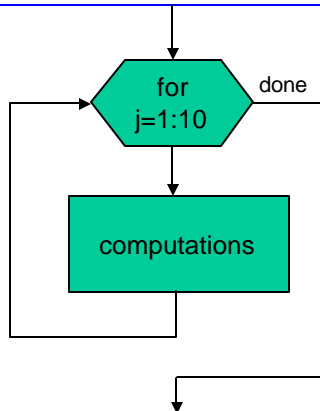
– Develop the process:

- Calculate the area
- Calculate the circumference
- If the area is big enough,
 - Display radius, area and circumference
 - otherwise, display nothing
- Repeat 10 times



4

FOR loop



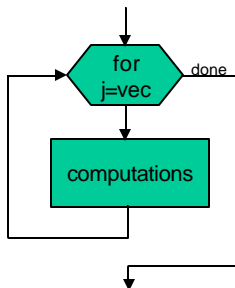
```
for j=1:10
% computations;
end
```

- Repeats for **specified** number of times
- **ALWAYS** executes computation loop at least once!!!
- Can use + or – increments
- Can escape (**BREAK**) out of computational loop before normal termination would occur.

5

More Details of the FOR Loop

```
x=[1.1,2.6,-3.4];
for x
% computations;
end
```



- **FOR** variable can be a vector of integer or floating point values
- Indexing is by columns only!

```
z=rand(4,5);
for z
% computations;
end
```

On your own:
What happens if the **FOR** variable is set to 0? to null (empty)? to NaN?

- **FOR** loop will execute for the 4 successive columns of z
- Each column of z contains a vector of 5 random values

6

Problem 1: ... cont'd

- **Step 4: Develop the solution**

- Write Matlab code

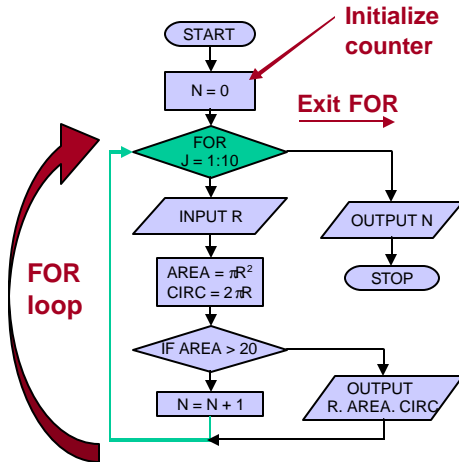
Use a **FOR** loop:

calculate the area and the circumference of **TEN** circles

allow the radius to be an input variable

output radius, area and circumference **IF** the area is greater than 20 square units.

output the number of circles with area \leq 20.



7

Problem 2: ... cont'd

```

% calculate the area and circumference of 10 circles
% print it out if the area is greater than 20
N = 0;
for i = 1:10
    radius = input( '\nPlease enter a radius:' );
    radius = abs(radius); % make sure always is positive!
    area = pi * radius ^2;
    circumference = 2 * pi * radius;
    if area > 20
        fprintf('\n Radius = %f units', radius);
        fprintf('\n Area = %f units squared', area);
        fprintf('\n Circumference = %f units', circumference);
    else
        N = N + 1;
    end
end
fprintf('\n %f circles with area less than 20', N);
  
```

Annotations in the image:

- Loop for i=1...10 (points to the `for i = 1:10` line)
- Print only if area>20 (points to the `if area > 20` line)
- End of FOR loop (points to the `end` line)

8

Problem 1 ... contd.

- **Step 5: Test the results**

```
>> basicFOR
Please enter a radius:0
Please enter a radius:1
Please enter a radius:2
Please enter a radius:3
Radius = 3.000000 units
Area = 28.274334 units squared
Circumference = 18.849556 units
Please enter a radius:-4
Radius = 4.000000 units
Area = 50.265482 units squared
Circumference = 25.132741 uni
Please enter a radius:5
Radius = 5.000000 units
Area = 78.539816 units squared
Circumference = 31.415927 units
Please enter a radius:6
Radius = 6.000000 units
Area = 113.097336 units squared
Circumference = 37.699112 units
Please enter a radius:7
Radius = 7.000000 units
Area = 153.938040 units squared
Circumference = 43.982297 units
Please enter a radius:8
Radius = 8.000000 units
Area = 201.061930 units squared
Circumference = 50.265482 units
Please enter a radius:9
Radius = 9.000000 units
Area = 254.469005 units squared
Circumference = 56.548668 units
3.000000 circles with area less than 20
>>
```

Handled
negative
radius

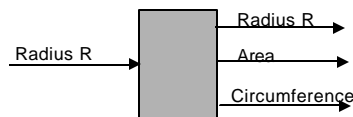
9

Problem 2: Introducing the **WHILE** Structure

- **Step 1: State the problem:** Write a MatLab program to
 - calculate the area and the circumference of **ANY NUMBER** of circles so long as the radius is greater than zero
 - take the radius as an input variable
 - output radius, area and circumference **IF** the area is greater than 20 square units.
 - output the number of circles with area \leq 20.

Step 2: Describe input and output

INPUT CALCULATE OUTPUT



How will we be able to stop the calculation process?
ANSWER : we'll stop when a zero radius is entered.

Step 3: Define test cases

R=0 → Area=0 and Circumference=0

R=1 → Area= π and Circumference= 2π

Create case with 0 areas $>$ 20 and with specified # $>$ 20 to test

10

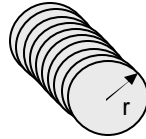
Problem 2 ... contd.

- **Step 4: Develop the solution**

- **Describe the algorithm:**

$$\text{Area} = \pi r^2$$

$$\text{Circum} = 2\pi r$$



- **Develop the process:**

- Repeat until radius = 0
 - calculate the circumference
 - calculate the area
 - If the area is big enough,
 - » Display radius, area and circumference
 - » otherwise, display nothing

11

Problem 2: ... cont'd

- **Step 4: Develop the solution**

- **Develop a Matlab solution**

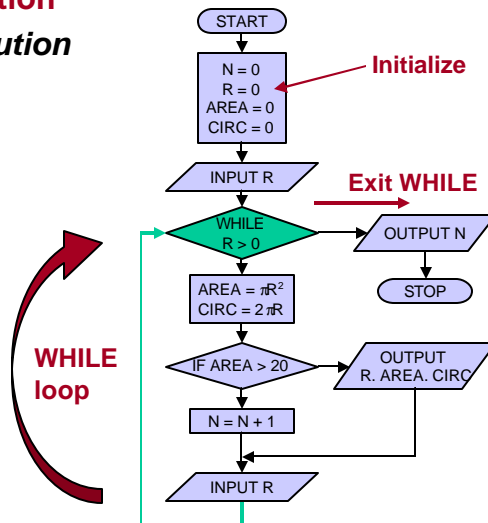
Use a **WHILE** loop:

calculate the area and the circumference of circles **while** the radius is > 0

allow the radius to be an input variable

output radius, area and circumference **IF** the area is greater than 20 square units.

output the number of circles with area \leq 20.



12

Problem 2 - Programming the **WHILE** Loop

```
% calculate the area and circumference of circles
% print results if the area is greater than 20; print the
% number of circles with area less than 20; terminate on area<=0
N = 0;
radius = input('\nPlease enter a radius: ');
while radius > 0
    area = pi * radius ^2;
    circumference = 2 * pi * radius;
    if area > 20
        fprintf('\n Radius = %f units', radius);
        fprintf('\n Area = %f units squared', area);
        fprintf('\n Circumference = %f units', circumference);
    else
        N = N + 1;
    end
    radius = input('\nPlease enter a radius: ');
end
fprintf('\n %f circles with area less than 20', N);
```

Loop while radius > 0

Note the need to repeat the user input here

While end

13

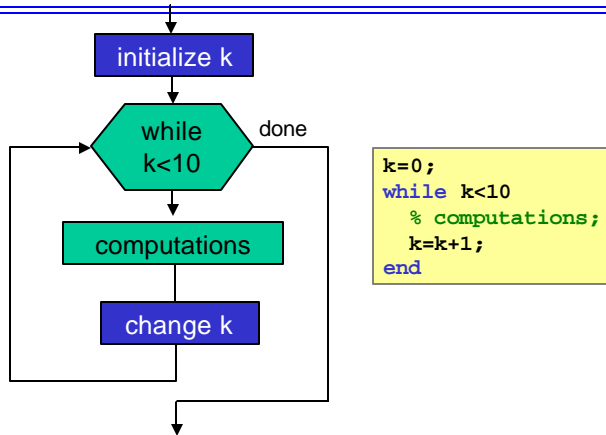
Problem 2 ... contd.

- **Step 5: Test the result**

```
Please enter a radius: 5
Radius = 5.000000 units
Area = 78.539816 units squared
Circumference = 31.415927 units
Please enter a radius: 4
Radius = 4.000000 units
Area = 50.265482 units squared
Circumference = 25.132741 units
Please enter a radius: 3
Radius = 3.000000 units
Area = 28.274334 units squared
Circumference = 18.849556 units
Please enter a radius: 2
Please enter a radius: 1
Please enter a radius: 0
2.000000 circles with area less than 20
>>
```

14

WHILE Loop



- Will do computational loop **ONLY** if **WHILE** condition is met
- Be careful to initialize **WHILE** variable
- Can loop forever if **WHILE** variable is not updated within loop!!!

15

Problem Solving

- **Linear Interpolation** is the process of calculating a straight line between two points and using this to compute intermediate results.
- If the points are close enough together, the errors can be quite small and the computations are quick and easy.
- Consider linear interpolation between 2 points on the X-Y plane represented as (x_1, y_1) and (x_2, y_2)
 - *Develop your own m-function to take the two points as inputs and the x value of the intermediate point to compute.*
 - *Return the computed y value*
 - *Test this for some data*

16

Summary

- **Topics Covered**

- `for` loop statement
- `while` loop statement
- **Summary**

- **Action Items**

- Review the lecture
- Keep exploring MATLAB!
- Come prepared to ask questions about MATLAB

17

Questions?

18