

CS 101 Computer Programming and Utilization

Lecture 6

For Loop and Arrays

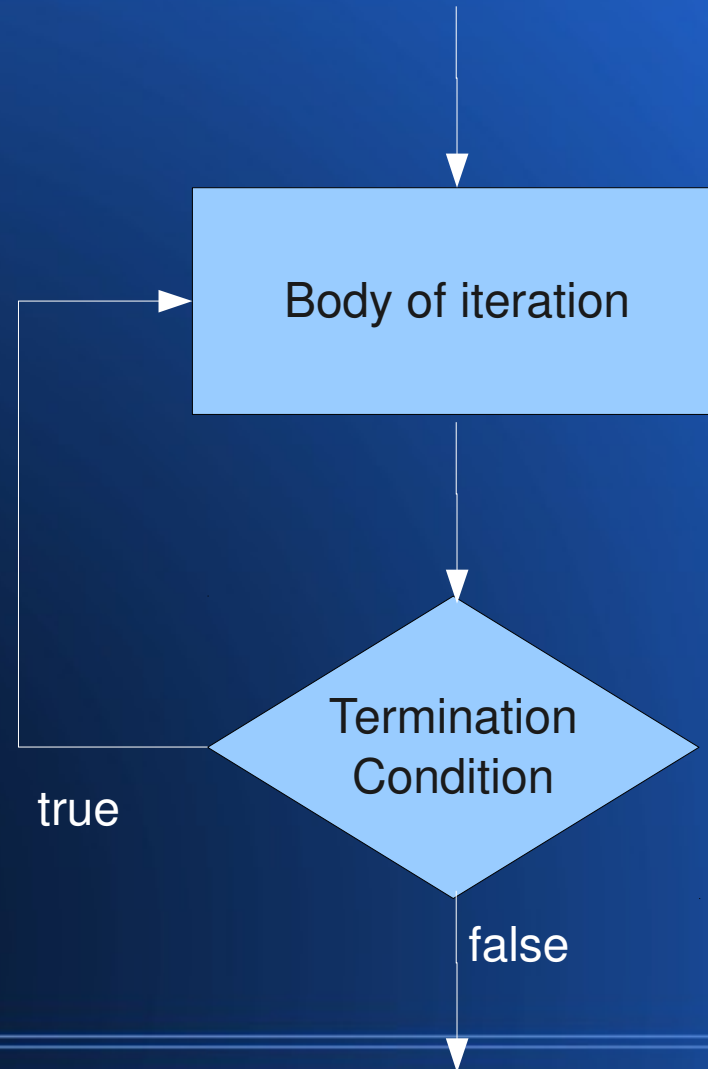
Jan 21 Friday 11:05-12:30 PCS D2
Jan 25 Tuesday 2:00-3:30 FCK D4

Prof. R K Joshi
Computer Science and Engineering
IIT Bombay
Email: rkj@cse.iitb.ac.in

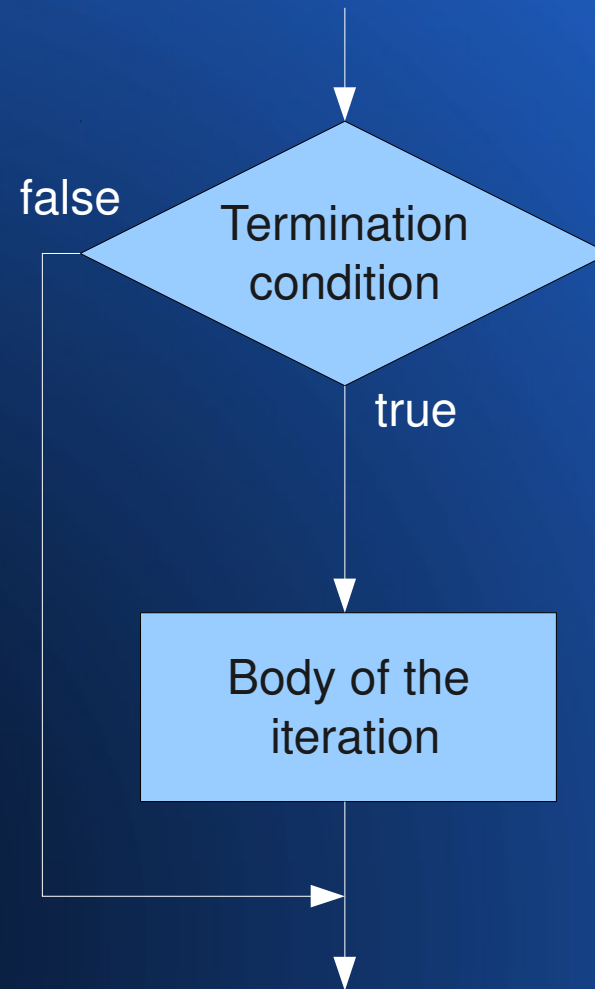
Revision: Iteration

- Iteration is repetition of something again and again
 - Iteration has a body
 - It has a terminating condition
 - In every iteration, both are executed exactly once
 - Either body before the terminating condition
 - Or the terminating condition before the body
 - Iteration terminates when terminating condition evaluates to value true
- `do {...;...;...;} while (c);`
 - `while (c) {...;...;...};`
 - Which one to choose?
 - Avoid redundant code
 - Avoid exit from within the body. We want exit from the terminating condition
 - Watch out trivial looping
 - Avoid extra assignments and extra checks

Control Flow Diagram for do while statement



Control Flow Diagram for while do statement



Loop/Iteration index

- An integer variable that keeps track of the number of times the iteration is executed
- It is initialized at the start of the iteration
- It is incremented in every iteration
- Such a variable is called loop index
- Or also called iteration index
 - demo

Counter

- An integer variable that counts something is called a counter
- It is initialized and incremented whenever the counting situation arises
- A loop index is a counter since it counts the number of iterations
- A variable that counts the no. of odd integers within range 0..9 is also a counter.

– demo

Flag

- A variable that indicates the boolean status of some situation occurring in the program
- It is initialized to appropriate value (true or false)
- A flag is set as soon as the situation is detected
 - Thus an occurrence of the situation is remembered
- Then it is used later in the program
- We used a flag during last lecture in one of our attempts at refining the while loop
 - Back to old demo

Collections of values

- `Int a,b,c;`
- `Char ch;`
- `Float r1, r2;`
- `Double d;`
 - All these are single values
- What if you wanted to store a collection of many values of one type?
- e.g. 10 integers corresponding to marks of 10 students?
- Using 10 variables is cumbersome
 - Not only many variables
 - Size of the control flow increases
 - You have observed this in lab 2

Arrays

- `int A[5]`
 - An array of 5 integers
 - How to get to the individual integers?
 - They are:
 - `A[0]`
 - `A[1]`
 - `A[2]`
 - `A[3]`
 - `A[4]`
 - Each one of numbers 0 to 4 has been used as an array index

Array Index

- We can also use a variable in place of a number

```
int A[10];  
  
i=0;  
A[i]=0;  
  
i=1;  
A[i]=1;  
  
i=2;  
A[i]=2;  
  
// and so on
```

Operations on Collections by Using Arrays inside iterations

- Use iteration index as array index
- An indexed value from an array can be used on left hand side or on the right hand side as lvalue or as rvalue respectively.

```
int i;  
  
i=0;  
  
while (i<n) {  
    A[i] = i;    // lvalue  
    cout << A[i]; // rvalue  
    i = i + 1;  
}
```

...

For loop

A simple way to right loops

Loop index initialization, loop termination in terms of loop index and the loop index step all at once place

```
for (i=0; i<n; i++) {  
  
}
```

demos