



C PROGRAMING

Ketan Kore

Sunbeam Infotech



Control Statements

- Decision or Selection
 - if-else
 - switch-case
- Iteration (loop)
 - for
 - while
 - do-while
- Jump
 - break
 - continue
 - goto
 - return



Loops

- Control statements used for repeating a set of instructions number of times is called as “LOOP”.
- Every loop has
 - Initialization statement
 - Terminating condition
 - Modification statement(Increment/Decrement)
 - Body of loop
- The variable that is used for terminating condition is referred as ‘loop variable’.



while loop

- Used to repeat a statement (or block) while an expression is true (not zero).

- **Syntax:**

```
initialization;  
while(condition) {  
    statement1;  
    statement2;  
    modification;  
}
```



for loop

- Used to repeat a statement (or block) while an expression is true (not zero).

- Syntax:

```
for(initialization; condition; modification) {  
    statement1;  
    statement2;  
}
```



do-while loop

- Used to repeat a statement (or block) while an expression is true (not zero).

- Syntax:

```
do {  
    statement1;  
    statement2;  
} while(condition);
```

- do-while is exit control loop.
- while & for are entry control loops.
- do-while is executed at least once.



Infinite loop

- If loop condition is always true, program never terminates.

```
while(1) {
```

```
    ...
```

```
}
```

```
for(;;) {
```

```
    ...
```

```
}
```

```
do {
```

```
    ...
```

```
} while(1);
```



break/continue

- break statement
 - Used to early exit from loop, or to exit an infinite loop
 - Takes control out of current loop and continues execution of statements after the loop.
 - Statements after break are skipped.
- continue statement
 - Used to continue next iteration of the loop.
 - Statements after continue are skipped (for current iteration).
- break is used with loop/switch case.
- continue used with only loop.
- In case of nested loops, break/continue affects current loop only (not outer).



goto statement

- Jumps to statement label, **must be within same function as the goto.**
 - Statement label is an identifier followed by a colon (:)
 - Unstructured control statement
 - Used rarely (less readable)
 - Advised to use only for forward jump
- Best use is to exit from deeply nested loops.
- Syntax:

```
goto label_name;  
..  
..  
label_name: C-statements
```





Thank you!

Ketan Kore<ketan.kore@sunbeaminfo.com>

