The University of Nottingham

UNITED KINGDOM · CHINA · MALAYSIA

Week 2- lecture 3

Control Statements - Iteration

Edited by: Dr. Saeid Pourroostaei Ardakani

Autumn 2021

Smile More®

University of Nottingham
UK | CHINA | MALAYSIA

Quiz!

What is the result of A below?

A = 1+3*4-(4/2)+2*3+5

A) 19

B) 0.

C) 22.

D) 29.

Quiz!

What is the result of A below?

A = 1+3*4-(4/2)+2*3+5

A) 19

B) 0.

C) 22.

D) 29.

Run the following code, What is the output?

```
1  // Fig. 3.13: fig03_13.c
2  // Preincrementing and postincrementing.
3  #include <stdio.h>
4
5  // function main begins program execution
6  int main( void )
7  {
8     int c; // define variable
9
10    // demonstrate postincrement
11    c = 5; // assign 5 to c
12    printf( "%d\n", c ); // print 5
13    printf( "%d\n", c++ ); // print 5 then postincrement
14    printf( "%d\n\n", c ); // print 6
15
16    // demonstrate preincrement
17    c = 5; // assign 5 to c
18    printf( "%d\n", c ); // print 5
19    printf( "%d\n", ++c ); // preincrement then print 6
20    printf( "%d\n", c ); // print 6
21 } // end function main
```

Overview

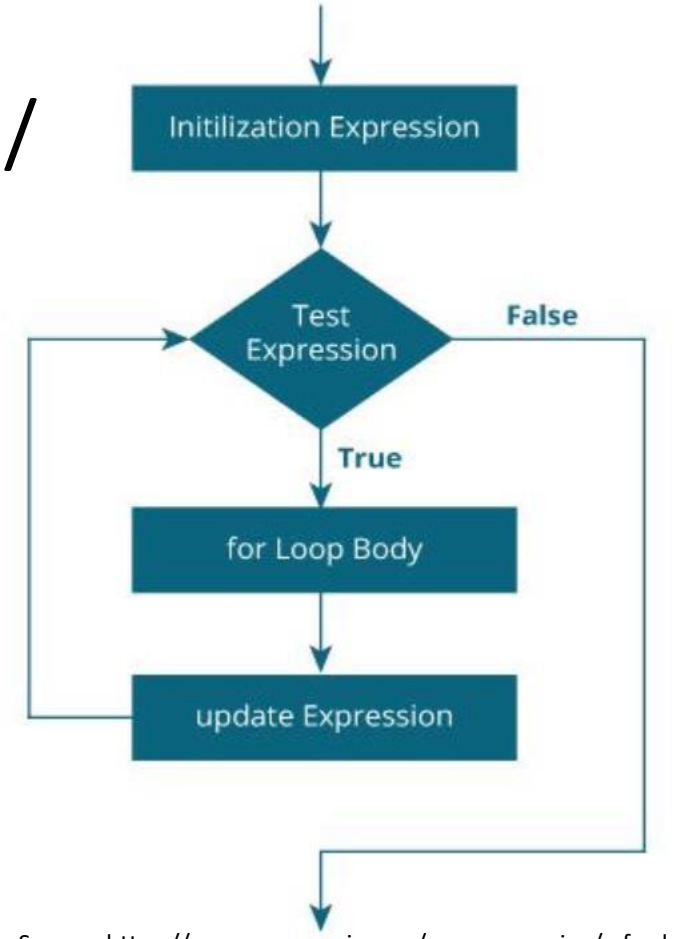
- **Control Structure in C**
- Return value of scanf
- break and continue

Control Structure in C

- Sequence structure
- Selection structure
  - if
  - if … else …
  - switch
- Iteration structure
  - while
  - do … while …
  - for

for Loop in C

- for(initialisation; condition; update)
  {
  /* statements */
  }



Source: https://www.programiz.com/c-programming/c-for-loop

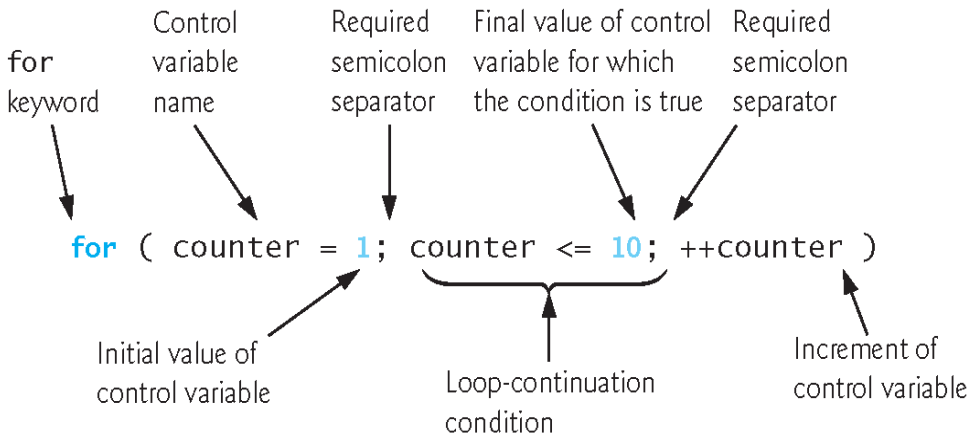
for Loop in C (2)

- The general format of the `for` statement is

```
for ( expression1; expression2; expression3 ) {
 statement
  }
```

where

- *expression1* initializes the loop-control variable
- *expression2* is the loop-continuation condition
- *expression3* increments the control variable.



Example: for Loop

```c
#include <stdio.h>

int main(void)
{
    int input = 0;
    int sum = 0;

    printf("Please enter an input: ");
    scanf("%d", &input);

    sum = sum + input;

    printf("Please enter an input: ");
    scanf("%d", &input);

    sum = sum + input;

    printf("The sum of two numbers is %d\n", sum);

    return 0;
}
```

```c
#include <stdio.h>

int main(void)
{
    int counter = 0;
    int input = 0;
    int sum = 0;

    for(counter = 0; counter < 2; counter++)
    {
        printf("Please enter an input: ");
        scanf("%d", &input);

        sum = sum + input;
    }

    printf("The sum of two numbers is %d\n", sum);

    return 0;
}
```

Example: for Loop (2)

The following examples show methods of varying the control variable in a `for` statement.

Vary the control variable from 1 to 100 in increments of 1.

```
for ( i = 1; i <= 100; ++ i )
```

Vary the control variable from 100 to 1 in increments of -1 (decrements of 1).

```
for ( i = 100; i >= 1; --i )
```

Vary the control variable from 7 to 77 in steps of 7.

```
for ( i = 7; i <= 77; i += 7 )
```

Vary the control variable from 20 to 2 in steps of -2.

```
for ( i = 20; i >= 2; i -= 2 )
```

Vary the control variable over the following sequence of values: 2, 5, 8, 11, 14, 17.

```
for ( j = 2; j <= 17; j += 3 )
```

Vary the control variable over the following sequence of values: 44, 33, 22, 11, 0.

```
for ( j = 44; j >= 0; j -= 11 )
```

Take Home exercise.

Consider the following problem statement:

A person invests $1000.00 in a savings account yielding 5% interest. Assuming that all interest is left on deposit in the account, calculate and print the amount of money in the account at the end of each year for 10 years. Use the following formula for determining these amounts:

$$a = p(1 + r)^n$$

where

p is the original amount invested (i.e., the principal)

r is the annual interest rate

n is the number of years

a is the amount on deposit at the end of the n^{th} year.

This problem involves a loop that performs the indicated calculation for each of the 10 years the money remains on deposit.

Take Home exercise.

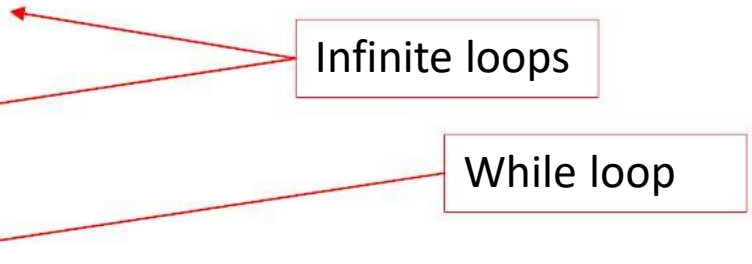
```
1  // Fig. 4.6: fig04_06.c
2  // Calculating compound interest.
3  #include <stdio.h>
4  #include <math.h>
5
6  // function main begins program execution
7  int main( void )
8  {
9     double amount; // amount on deposit
10    double principal = 1000.0; // starting principal
11    double rate = .05; // annual interest rate
12    unsigned int year; // year counter
13
14    // output table column heads
15    printf( "%4s%21s\n", "Year", "Amount on deposit" );
16
17    // calculate amount on deposit for each of ten years
18    for ( year = 1; year <= 10; ++year ) {
19
20       // calculate new amount for specified year
21       amount = principal * pow( 1.0 + rate, year );
22
23       // output one table row
24       printf( "%4u%21.2f\n", year, amount );
25    } // end for
26 } // end function main
```

Take Home exercise.

```
Year    Amount on deposit
   1              1050.00
   2              1102.50
   3              1157.63
   4              1215.51
   5              1276.28
   6              1340.10
   7              1407.10
   8              1477.46
   9              1551.33
  10              1628.89
```

Omitting Expressions

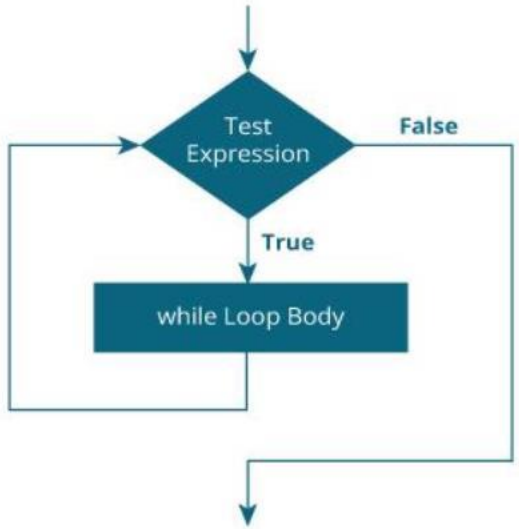
- int a = 0;
  for(; a < 5; a++) { … }
- for(a = 0; a < 5;){
  a++;
  }
- for(a = 0; ; a++) {… }
- for(; ;) { … }
- for(; a < 5;) { … }



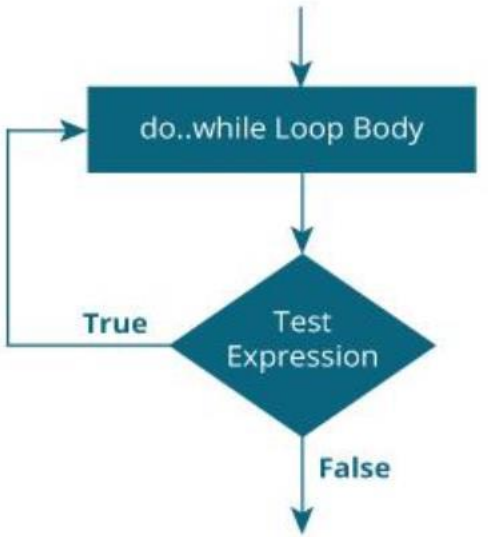
while and do-while Loop in C

- while(test)
  {
  /* statements */
  }

- do
  {
  /* statements */
  }
  while(test);



Source: https://www.programiz.com/c-programming/c-do-while-loops



Example: while and do-while Loop

```c
#include <stdio.h>

int main(void)
{
    int input = 0;
    int sum = 0;

    do
    {
        printf("Please enter an input: ");
        scanf("%d", &input);

        if(input > 0)
        {
            sum = sum + input;
        }

    }while(input > 0);

    printf("The sum of two numbers is %d\n", sum);

    return 0;
}
```

```c
#include <stdio.h>

int main(void)
{
    int counter = 0;
    int input = 0;
    int sum = 0;
    int limit = 0;

    printf("Please enter the number of input: ");
    scanf("%d", &limit);

    while(counter < limit)
    {
        printf("Please enter an input: ");
        scanf("%d", &input);

        sum = sum + input;
        counter = counter + 1;
    }

    printf("The sum of two numbers is %d\n", sum);

    return 0;
}
```

Example: Menu

```
#include <stdio.h>

int main(void)
{
    int keepGoing = 1;

    do
    {
        // Print the menu and get the user's choice.
        printf("1) Add\n");
        printf("2) Multiply\n");
        printf("3) Mod\n");
        printf("4) Quit\n");
        printf("Enter option: ");

        int option = 0;
        scanf("%d", &option);

        if(option == 1)
        {
            // Add 2 numbers.
            int n1 = 0;
            printf("Enter first number: ");
            scanf("%d", &n1);

            int n2 = 0;
            printf("Enter second number: ");
            scanf("%d", &n2);

            int ans = n1 + n2;
            printf("The answer is %d\n\n", ans);
        }
        else if(option == 2)
        {
            // Multiply 2 numbers.
            int n1 = 0;
            printf("Enter first number: ");
            scanf("%d", &n1);

            int n2 = 0;
            printf("Enter second number: ");
            scanf("%d", &n2);

            int ans = n1 * n2;
            printf("The answer is %d\n\n", ans);
        }
        else if(option == 3)
        {
            // Modulus
            int n1 = 0;
            printf("Enter first number: ");
            scanf("%d", &n1);

            int n2 = 0;
            printf("Enter second number: ");
            scanf("%d", &n2);

            int ans = n1 % n2;
            printf("The answer is %d\n\n", ans);
        }
        else if(option == 4)
        {
            // Quit
            keepGoing = 0;
        }
    }while(keepGoing);

    return 0;
}
```

Overview

- Control Structure in C
- **Return value of scanf**
- break and continue

Understanding Input Buffer

```c
#include <stdio.h>

int main(void)
{
    int input = 0;
    int sum = 0;

    printf("Please enter an input: ");
    scanf("%d", &input);

    sum = sum + input;

    printf("Please enter an input: ");
    scanf("%d", &input);

    sum = sum + input;

    printf("The sum of two numbers is %d\n", sum);

    return 0;
}
```

```
C:\Users\z2017233\Desktop>iteration
Please enter an input: 1
Please enter an input: 2
The sum of two numbers is 3

C:\Users\z2017233\Desktop>iteration
Please enter an input: 1 2
Please enter an input: The sum of two numbers is 3

C:\Users\z2017233\Desktop>iteration
Please enter an input: 1 2 3
Please enter an input: The sum of two numbers is 3

C:\Users\z2017233\Desktop>iteration
Please enter an input: 1 2 3 4
Please enter an input: The sum of two numbers is 3
```

- scanf will read from buffer without waiting for any user input if the buffer is **NOT** empty.

Return value of scanf()

- One way to validate user's input.

```
C:\Users\z2017233\Desktop>iteration
Please enter your date of birth (dd/mm/yyyy): 01/02/1234
There are 3 correctly formatted values
The values are 1, 2 and 1234

C:\Users\z2017233\Desktop>iteration
Please enter your date of birth (dd/mm/yyyy): 01 02 1234
There are 1 correctly formatted values
The values are 1, 0 and 0
```

```
181  #include <stdio.h>
182
183  int main(void)
184  {
185      int day = 0;
186      int month = 0;
187      int year = 0;
188      int count = 0;
189
190      printf("Please enter your date of birth (dd/mm/yyyy): ");
191      count = scanf("%d/%d/%d", &day, &month, &year);
192
193      printf("There are %d correctly formatted values\n", count);
194      printf("The values are %d, %d and %d\n", day, month, year);
195
196
197      return 0;
198  }
```

Overview

- Control Structure in C
- Return value of scanf
- **break and continue**

break vs continue Statement

- The **break** statement, when executed in a loop (e.g., while), causes an immediate exit from that statement. Program execution continues with the next statement

- The **continue** statement, when executed in loop, skips the remaining statements in the body of that control statement and performs the next iteration of the loop.

break vs continue Statement (2)

Completely terminates a loop

```
int i;
for(i = 1; i < 10; i++){
    if(i == 5)
        break;
    printf("%d ", i);
}
printf("End = %d\n", i);
```

Displays 1 2 3 4 End = 5

Terminates the **current iteration** of a loop

```
int i;
for(i = 1; i < 10; i++){
    if(i < 5)
        continue;
    printf("%d ", i);
}
```

Displays 5 6 7 8 9

break vs continue Statement (3)

```
while (testExpression) {
   // codes
   if (condition to break) {
      break;
   }
   // codes
}
```

```
do {
   // codes
   if (condition to break) {
      break;
   }
   // codes
}
while (testExpression);
```

Note the directions of the arrows!!

```
for (init; testExpression; update) {
   // codes
   if (condition to break) {
      break;
   }
   // codes
}
```

Source: https://www.programiz.com/c-programming/c-break-continue-statement

```
while (testExpression) {
   // codes
   if (testExpression) {
      continue;
   }
   // codes
}
```

```
do {
   // codes
   if (testExpression) {
      continue;
   }
   // codes
}
while (testExpression);
```

```
for (init; testExpression; update) {
   // codes
   if (testExpression) {
      continue;
   }
   // codes
}
```

Using break

- In this example, watch out for integer division.

```
122  #include <stdio.h>
123
124  int main(void)
125  {
126      int input = 0;
127      int sum = 0;
128      int count = 0;
129
130      do
131      {
132          printf("Please enter an input (-1 to quit): ");
133          scanf("%d", &input);
134
135          if(input == -1)
136          {
137              break;
138          }
139          sum = sum + input;
140          count++;
141
142      }while(input != -1);
143
144      printf("The sum of the input numbers is %d\n", sum);
145      printf("The average of the input numbers is %d\n", (sum / count));
146
147      return 0;
148  }
```

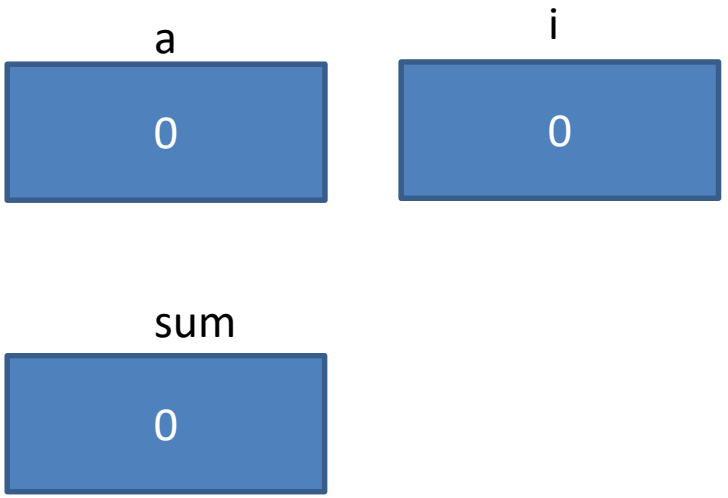
Summary

- Control Structure in C
- Return value of scanf
- break and continue

Quiz!

Complete and run the code. What are the values in the blue boxes?

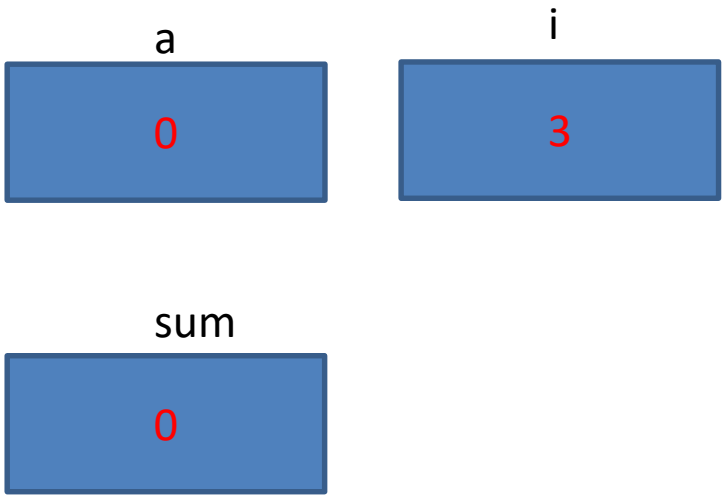
```
int a= 0;
sum = 0;
For (i=0; i<3; i++){
        if (a== 0){
                sum = sum+i;
        }
        Else{
                a=0;
                continue;
                sum = sum*1;
        }
}
```



Quiz!

Complete and run the code. What are the values in the blue boxes?

```
int a= 0;
sum = 0;
For (i=0; i<3; i++){
        if (a== 0){
                sum = sum+i;
        }
        Else{
                a=0;
                continue;
                sum = sum*1;
        }
}
```



Complete and run the following code, what is the output?

```
int main (void){

        unsigned int x;
        for (x = 1; x <= 10; x++)
        {
                if (x ==5)
                {
                        continue;
                }
                printf("%u", x);
        }
printf("\n continue used to skip");
}
```

Complete and run the following code, what is the output?

```
int main (void){

        unsigned int x;
        for (x = 1; x <= 10; x++)
        {
                if (x ==5)
                {
                        continue;
                }
                printf("%u", x);
        }
printf("\n continue used to skip");
}
```

Output:

1 2 3 4 6 7 8 9 10
continue used to skip