# Conditional Statements - C1W3

**T1** Chapters: **5-13**

## Boolean Expressions

* Boolean refers to the values corresponding to true and false in the C++ programming language.
* Boolean expressions are those expressions that result in a true or false.
* To compose such expressions the operators available are as follows
  + !, && and || for logical NOT, AND and OR respectively.
  + <, > for Less Than and Greater Than respectively.
  + <=, >= for Less Than Or Equal To and Greater Than Or Equal To respectively.
  + ==, != for Equal and Not Equal respectively.
* For example,
* int is\_even = (4 % 2) != 0;
* the value of is\_even is going to be false, since the arithmetic expression 4 % 2 produces a 0.
* OR
* int is\_true = (4 > 0) && (2 < 3);
* the value of is\_true is going to be true, since the boolean expressions 4 > 0 and 2 < 3 are both true. If either would have been false, the value of is\_true would be false.

## Exercises

1. Show the output produced by each of the following code snippets. Assume that i, j and k are int variables.
   1. // E  
      i = 2;  
      j = 3;  
      k = i \* j == 6;  
      printf("%d", k);
   2. // M  
      i = 5;  
      j = 10;  
      k = 1;  
      printf("%d", k > i < j);
   3. // H  
      i = 3;  
      j = 4;  
      k = 5;  
      printf("%d", i % j + i < k);
   4. // H  
      i = 3;  
      j = 4;  
      k = 5;  
      printf("%d", i < j || ++j < k);  
      printf("%d %d %d", i, j, k);

## if

* The if statement has the following structure,
* if (EXPRESSION)  
  {  
   BODY  
  }
* The if statement executes code in the BODY only when the boolean EXPRESSION evaluates to true.
* For example,
* #include <cstdio>
    
    
  int main()
    
  {
    
   int x = 5;
    
    
   if ((x > 0) && (x != 1))
    
   {
    
   printf("Y\n");
    
   }
    
    
   if (int x = 5; (x > 0) && (x != 1))
    
   {
    
   printf("Y\n");
    
   }
    
  }
* In the program above, given that x has the value 5, the if statement evaluates the boolean expression (x > 0) && (x != 1) to true and thus displays Y to the console.

## else

An if statement can have an else block if an alternative piece of code needs to be executed when the boolean expression inside if evaluates to false. It has the structure,

if (EXPRESSION)  
{  
 BODY\_1  
}  
else  
{  
 BODY\_2  
}

Here, BODY\_2 executes only when EXPRESSION evaluates to false.

For example,

#include <cstdio>
  
  
int main()
  
{
  
 int x = 5;
  
  
 if ((x < 0) && (x != 1))
  
 {
  
 printf("Y\n");
  
 }
  
 else
  
 {
  
 printf("N\n");
  
 }
  
}

In the program above, given that x has the value 5, the if statement evaluates the condition (x < 0) && (x != 1) to false. Then, the code printf function inside the else block is executed, displaying N on the console.

## else if

When a program requires several alternative conditions or boolean expressions to evaluate and execute code when one of these expressions evaluate to True, an if-else-if chain could be employed. It has the following structure,

if (EXPRESSION\_1)  
{  
 BODY\_1  
}  
else if (EXPRESSION\_2)  
{  
 BODY\_2  
}  
...  
else if (EXPRESSION\_N\_1)  
{  
 BODY\_N\_1  
}  
else  
{  
 BODY\_N  
}

For example,

#include <cstdio>
  
  
int main()
  
{
  
 int x = 0;
  
 printf("Enter an integer: ");
  
 scanf("%d", &x);
  
  
 if ((x % 2) == 0)
  
 {
  
 printf("Even\n");
  
 }
  
 else if ((x % 2) != 0)
  
 {
  
 printf("Odd\n");
  
 }
  
 else if ((x % 5) == 0)
  
 {
  
 printf("Divisible by 5\n");
  
 }
  
 else
  
 {
  
 printf("x = %d\n", x);
  
 }
  
}

## Exercises

1. Rewrite the above program using nested if-else statements. (E)
2. When stocks are sold or purchased through a broker, the broker's commission is often computed using a sliding scale that depends upon the value of the stocks traded. Given that a broker charges the amounts shown in the following table:

| * Transaction Size | * Commission Rate |
| --- | --- |
| * Under £2,500 | * £30 + 1.7% |
| * £2,500 - £6,250 | * £56 + 0.66% |
| * £6,250 - £20,000 | * £76 + 0.34% |
| * £20,000 - £50,000 | * £100 + 0.22% |
| * £50,000 - £500,000 | * £155 + 0.11% |
| * Over £500,000 | * £255 + 0.09% |

* The minimum charge is £39. Implement a program that asks the user to enter the amount or value of the trade, then displays the amount of the commission calculated using the table above. (M)

1. Modify the above program by making both of the following changes: (H)
   1. Ask the user to enter the number of shares and the price per share, instead of the value of the trade.
   2. Add statements that compute the commission charged by a rival broker (£33 plus 3p per share for fewer than 2000 shares; £33 plus 2p per share for 2000 shares or more). Display the rival's commission as well as the commission charged by the original broker.

## switch

A switch statement is used when a piece of code is to be selected for execution depending on the value of an expression. It has the following structure,

switch (EXPRESSION)  
{  
 case VALUE\_1:  
 BODY\_1  
 break;  
 case VALUE\_2:  
 BODY\_2  
 break;  
 ...  
 case VALUE\_N:  
 BODY\_3  
 break;  
 default:  
}

For example,

#include <cstdio>
  
  
int main()
  
{
  
 const int RED = 0;
  
 const int GREEN = 1;
  
 const int BLUE = 2;
  
  
 printf("Press 0 for RED\n");
  
 printf("Press 1 for GREEN\n");
  
 printf("Press 2 for BLUE\n");
  
  
 int opt = -1;
  
 scanf("%d", &opt);
  
  
 switch (opt)
  
 {
  
 case RED:
  
 printf("RED\n");
  
 break;
  
 case GREEN:
  
 printf("GREEN\n");
  
 break;
  
 case BLUE:
  
 printf("BLUE\n");
  
 break;
  
 default:
  
 printf("Invalid option\n");
  
 return -1;
  
 }
  
  
 return 0;
  
}

**NOTE** An if-else if chain is equivalent to a switch statement of the above kind.

## Exercises

1. Using a switch statement, write a program that converts a numerical grade into a letter grade:

* Enter a numerical grade: 84  
  Letter grade: B
* Use the following grading scale:

| * GRADE | * MARKS |
| --- | --- |
| * A | * 90-100 |
| * B | * 80-89 |
| * C | * 70-79 |
| * D | * 60-69 |
| * F | * 0-59 |

* Print an error message if the grade is larger than 100 or less than 0. (M)

1. Write a program that asks the user for a two-digit number, then prints the English word for the number. (M)

* Enter a two-digit number: 45  
  You entered the number forty-five.

# Homework

1. **T1** Chapters:
   1. **14-16**
   2. **20-23**
2. Using a while loop, write a program that prints numbers in increasing order from 0 to 9. (E)
3. Using a while loop, write a program that prints numbers in decreasing order from 9 to 0. (E)
4. Using a while loop, write a program that asks the user to enter a number and breaks out of the loop if it is divisible by a prime number under 10. (M)