

Tutorial 5

Md. Tanvir Hassan

Email: md.hassan@mail.concordia.ca

Do/While Loop

Syntax:

```
do {  
    // code block to be executed  
}  
while (condition);
```

Example:

```
int i = 0;  
do {  
    cout << i << "\n";  
    i++;  
}  
while (i < 5);
```

Nested Loop

Syntax:

```
for ( init; condition; increment )  
{  
    for ( init; condition; increment )  
    {  
        // code block to be executed  
    }  
  
    // code block to be executed  
}
```

Example:

```
for(int i=0; i<5; i++)  
{  
    for(int j = 0; j<=i; j++)  
    {  
        cout <<"*";  
    }  
    cout <<endl;  
}
```

Brute Force Algorithm

- straightforward methods of solving a problem that rely on sheer computing power and trying every possibility rather than advanced techniques to improve efficiency



Exercise 1

(Pythagorean Triples) A right triangle can have sides that are all integers. A set of three integer values for the sides of a right triangle is called a Pythagorean triple. These three sides must satisfy the relationship that the sum of the squares of two of the sides is equal to the square of the hypotenuse. Find all Pythagorean triples for side1, side2 and hypotenuse all no larger than 500. Use a triple-nested for loop that tries all possibilities. This is an example of brute force computing. You'll learn in more advanced computer science courses that there are many interesting problems for which there's no known algorithmic approach other than sheer brute force.

Exercise 2

(*De Morgan's Laws*) In this chapter, we discussed the logical operators `&&`, `||` and `!`. De Morgan's laws can sometimes make it more convenient for us to express a logical expression. These laws state that the expression `!(condition1 && condition2)` is logically equivalent to the expression `(!condition1 || !condition2)`. Also, the expression `!(condition1 || condition2)` is logically equivalent to the expression `(!condition1 && !condition2)`. Use De Morgan's laws to write equivalent expressions for each of the following, then write a program to show that the original expression and the new expression in each case are equivalent:

a) `!(x < 5) && !(y >= 7)`