

## Summation

- **Problem:**

Write a C++ program to find the sum of the squares of the integers from 1 to 30. Print the sum of the squares.

$$\text{i.e., } \text{sum} = 1^2 + 2^2 + 3^2 + \dots + 29^2 + 30^2$$

- **Program:**

```
/* program 4
 * find the sum of the first 30 squares
 */
#include <iostream>
using namespace std;

int main()
{
    int i,sum;                // variables

    sum = 0;                  // initialize sum
    /* recursive addition of squares */
    for (i = 1; i <= 30; i++)
        sum = sum + (i * i);
    cout << sum << " is the sum of the first 30 squares."
         << endl;
    return 0;
}
```

## Declaring a Loop Index

- In the above program, the variable "i" was used solely within the for loop as a loop index. In C++ it is customary to declare such variables directly within the loop header. A variable so declared, has no meaning anywhere in the program, except within the loop in which it was declared.

- **Example:**

```
/* program 4
 * find the sum of the first 30 squares
 */
#include <iostream>
using namespace std;

int main()
{
    int sum = 0;                                // variables

    /* recursive addition of squares */
    for (int i = 1; i <= 30; i++)
        sum += (i * i);
    cout << sum << " is the sum of the first 30 squares."
         << endl;
    return 0;
}
```

## Defining and Using Constants

- **Defining Constants (The const statement):**

***const datatype identifier = constant\_value;***

- **Using Constants:**

```
/* program 4
 * find the sum of the first NUMBERTOSUM squares
 */
#include <iostream>
using namespace std;

const int NUMBERTOSUM = 30; //constant definition

int main()
{
    int sum = 0; // variables

    for (int i = 1; i <= NUMBERTOSUM; i++)
        sum += i * i; //recursive addition of squares
    cout << sum << " is the sum of the first " <<
        NUMBERTOSUM << " squares." << endl;
    return 0;
}
```

- **Note:** To change the number of integers we wish to sum, we only need to change the value of NUMBERTOSUM at its declaration.

- **Using a Value Entered at the Keyboard:**

```
/* program 4
 * find the sum of the first 'number_to_sum' squares
 * where number_to_sum is entered at the keyboard
 */
#include <iostream>
using namespace std;

int main()
{
    int sum = 0;                // variables
    int number_to_sum;

    cout << "Enter the number of squares to be summed: ";
    cin >> number_to_sum;

    for (int i = 1; i <= number_to_sum; i++)
        sum += (i * i);

    cout << sum << " is the sum of the first " <<
        number_to_sum << " squares." << endl;

    return 0;
}
```

## Nested for Loops

- **Problem:**

Write a program to produce a multiplication table that covers the integers 1 through 10.

- **Output:**

X	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
...										
10	10	20	30	40	50	60	70	80	90	100

- **Pseudocode:**

*print the headings at the top of the page  
construct each row of the multiplication table  
print the table*

- **Pseudocode Refinement:**

*print the headings at the top of the page  
for each multiplicand (m1) from 1 to 10  
    print a line of output showing m1 times each multiplier  
        (m2) from 1 to 10.*

- **Further Refinement:**

*print a line introducing the table  
print a heading line of multipliers from 1 to 10  
for each multiplicand (m1) from 1 to 10  
    start a new line of output by printing m1  
    for each multiplier (m2) from 1 to 10  
        print m1 \* m2 under the heading for m2*

● **Program:**

```
/* program to produce a multiplication table
 * for the integers 1 to 10
 */
#include <iostream>
using namespace std;

int main()
{
    cout << "\tThis is a Multiplication Table from 1 to 10"
         << endl << endl;
    cout.width(5);
    cout << "X";

    /* loop to print the heading of multipliers */
    for (int m2 = 1; m2 <= 10; m2++)
    {
        cout.width(5);
        cout << m2;
    }
    cout << endl;

    /* nested loop to print the table */
    for (int m1 = 1; m1 <= 10; m1++)
    {
        cout.width(5);
        cout << m1;           //prints the multiplicand
        for (int m2 = 1; m2 <= 10; m2++)
        {
            cout.width(5);
            cout << m1 * m2; //prints the product
        }
        cout << endl;
    }
    return 0;
}
```