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.

i.e., sum = 1^2 + 2^2 + 3^2 + ... + 29^2 + 30^2

● Program:
/* program 4
 * find the sum of the first 30 squares
 */
#include <stdio.h>
void main()
{
    int i,sum; // variables
    sum = 0; // initialize sum
    /* recursive addition of squares */
    for (i = 1; i <= 30; i + + )
        sum = sum + (i * i);
    printf("%d is the sum of the first 30 squares.\n",sum);
}
Defining and Using Constants

- **Defining Constants (The `#define` Directive):**
  
  ```
  #define identifier constant_value
  ```

- **Using Constants:**
  
  ```
  /* program 4
  * find the sum of the first NUMBERTOSUM squares
  */
  #include <stdio.h>
  #define NUMBERTOSUM 30
  void main()
  {
    int i,sum; // variables
    sum = 0; // initialize sum
    for (i = 1; i <= NUMBERTOSUM; i++)
      sum = sum + (i * i); //recursive addition of squares
    printf("%d is the sum of the first %d squares.\n", sum, NUMBERTOSUM);
  }
  ```

- **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 <stdio.h>
void main()
{
    int i,sum = 0; // variables
    int number_to_sum;

    printf("Enter the number of squares to be summed: ");
    scanf("%d", &number_to_sum);
    for (i = 1; i <= number_to_sum; i++)
        sum += (i * i);
    printf("%d is the sum of the first %d squares.\n", sum, number_to_sum);
}
Nested for Loops

● **Problem:**
  Write a program to produce a multiplication table that covers the integers 1 through 10.

● **Output:**

\[
\begin{array}{ccccccccccc}
X & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
2 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 \\
3 & 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 27 & 30 \\
... & & & & & & & & & & \\
10 & 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\
\end{array}
\]

● **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 <stdio.h>
void main()
{
    int m1,m2; //m1= multiplicand, m2= multiplier

    printf("This is a Multiplication Table from 1 to 10\n\n");
    printf(" X")
    /* loop to print the heading of multipliers */
    for (m2 = 1; m2 <= 10; m2++)
    {
        printf(" %5d",m2);
        printf("\n")
    }/* nested loop to print the table */
    for (m1 = 1; m1 <= 10; m1++)
    {
        printf(" %2d",m1); //prints the multiplicand
        for (m2 = 1; m2 <= 10; m2++)
        {
            printf(" %5d", m1 * m2);
            printf("\n")
        }
    }
}