

Solved problem

//this program illustrate how to define and Use a Function that has Arguments, The function requires two // arguments. The first argument is of type char and the second is of type int. This reflects the use of the // function in the pseudocode. The first argument that we shall pass to the function will be the character th // at we want the function to print. The second argument will be the number of times we want the function // to display the character.

```
#include <iostream>

using namespace std;

void Print_Char(char, int);

int main()
{
    int i,
        number_of_rows;
    char ch;

    cout << "Enter the character you want to display: ";
    cin >> ch;
    cout << endl;
    cout << "Enter the number of rows you want to display: ";
    cin >> number_of_rows;

    cout << endl;
    for (i = 1; i <= number_of_rows; ++i)
    {
        Print_Char(ch, i);
        cout << endl;
    }

    cout << endl;
    return 0;
}

void Print_Char(char display_char, int count)
{
    int j;
    for (j = 1; j <= count; ++j)
        cout << display_char;

    return;
}
```

Program Output

```
Enter the character you want to display: Z
Enter the number of rows you want to display: 5
Z
ZZ
ZZZ
ZZZZ
ZZZZZ
```

Exercises

- [1] Given the following function prototype, is there anything invalid about the use of the function in `main()`? If there is anything invalid, correct it.

```
int Func8(int, double);
int main()
{
    int i = 5,
        j;
    double db = 8.70;
    j = Func8(db, i);
    return 0;
}
```

- [2] Assume the following program outline.

```
int i1 = 9;
int main()
{
    int i2 = 5,          i3;          i3 = i1 + 7;
    i1 = Func12a(i1, i2);
    Func12b(i2, i3);
    return 0;
}
int i3 = 2;
int Func12a(int i1, int i3)
{
    static int i2 = 4;
    return i1 + i2 + i3;
}
Func12b(int j, int k)
{
    int i3 = 1;
    j = i3 + i2 + k;
}
```

1. Identify those variables that are local and those that are global.
2. State which assignment statements are valid and which are invalid.
3. Code a function `Rectangle_Area()` that takes two `double` arguments representing the length and width of a rectangle. The function should return the area ($\text{length} * \text{width}$) of the rectangle.
4. Code a function `Rectangle_Perim()` that takes two `double` arguments representing the length and width of a rectangle. The function should return the perimeter ($2 * \text{length} + 2 * \text{width}$) of the rectangle.
5. Code a function `Circle_Area()` that takes one `double` argument that represents the radius of a circle. The function should return the area ($3.14159 * \text{radius} * \text{radius}$) of the circle.
6. Code a function `Circle_Circum()` that takes one `double` argument that represents the radius of a circle. The function should return the circumference ($3.14159 * 2 * \text{radius}$) of the circle.
7. Code a function `Min()` that takes two `double` arguments and that returns the value of the smaller of the two arguments.
8. Code a function `Sgn()` that takes one `double` argument and returns +1 if the argument is positive; returns -1 if the argument is negative; and returns 0 if the argument is zero.
9. Code a function `Calc_Discount()` that has two `double` arguments representing the price of an item and a discount amount as a percent. The function should apply the discount to the price and return the discount amount.

Programming Problems

- [1] Write a program that asks the user to enter two decimal numbers. The program should calculate and display the product and quotient of the two numbers. Use a function `Product()` to calculate the product. The function should have two arguments, which are the two numbers that the user inputs. The function should return the product of the numbers. Use a function `Quotient()` that has two arguments, which are the numbers input by the user. The function should return the quotient of the first number divided by the second. If the second number is zero (recall that division by zero is not allowed), display an error message and exit the program.
- [2] Write a program that asks the user to enter two decimal numbers. The program should display the larger of the two numbers. Use a function `Max()` to find the larger number. `Max()` should have two arguments, which are the numbers that the user input. `Max()` should return the larger of the two numbers.
- [3] Write a program that raises an integer to a positive integer power. The program should prompt the user to enter the base and the exponent, and the program should display the corresponding power. Use a function `Power()` to calculate the power. Pass the base and the exponent to the function. `Power()` should calculate the power by multiplying base by itself exponent times. The function should return the value of the power.
- [4] The Big Racket Tennis Club wants you to write a program to calculate the yearly dues for its members. The program should prompt the user for the membership type (either 'I' for individual or 'F' for family) and the number of years the person has been a member. Use a function `Get_Member_Type()` that prompts the user for his or her membership type, obtains the type from the user and returns the type to `main()`. Use a function `Years()` that prompts the user for the number of years he or she has been a member and returns that number to `main()`.
 - Use a function `Dues()` to calculate the member's dues. Pass the member type and number of years to the function. Calculate the dues as follows: If the person is a family member and has been a club member for more than three years, the dues are \$2400.00. If the person is a family member and has been a club member for three years or less, the dues are \$3000.00. If the person is an individual member and has been a club member for more than five years, the dues are \$1500.00. If the person is an individual member and has been a club member for five years or less, the dues are \$1900.00. The function `Dues()` should return the dues to `main()`.
 - Use a function `Display_Member_Info()` to display the type of membership, the number of years the person has been a member, and the dues, all of which should be passed to the function as arguments.
- [5] A piece of equipment, such as a computer, loses value over time. The amount of this lost value is called **depreciation**. The simplest way to determine depreciation is by the **straight-line** method. If the asset has a useful life of n years, the straight-line method of depreciation assumes a depreciation of $1/n$ of the item's value each year. Assume a particular \$3,000 computer has a useful life of five years. At the end of the five years, assume it has a **scrap value** of \$500. Thus, the net cost of the computer is $3000 - 500 = \$2500$. Using the straight-line method, the computer depreciates \$500 ($= 2500 \times 1/5$) each of the five years of its useful life. After two years, the computer depreciates \$1,000. Thus, the **book value** of the computer after two years (the difference between the cost of the item and the depreciation to date) is \$2,000 ($= 3000 - 1000$).
 - Write a program that calculates the depreciation of an asset using the straight-line method. The program should ask the user for the initial cost of the item, the scrap value of the item, and the item's useful life in years. Use a function `Depreciation()` that calculates and displays a table that shows the book value of the item after each year of its useful life. The function should have three arguments: the initial cost, the scrap value, and the life of the item. The function should not return a value.