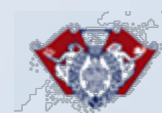


Practice 6



• Problem 1: Sum of integers using *for* statement

- Write a program that calculate the sum of integers.
- The program first get a user's integer input (2~100).
- Then, calculate the sum of integers from 1 to the integer entered. Use "for" statement.
- Display the result as follows.

```
Enter an integer: 4  
1 + 2 + 3 + 4 = 10
```

- Problem 2: Sum of integers using *while* statement

- Do the same calculation as the problem 1, but use “while” statement this time.

• Problem 3: Pattern printing

- Write a program that creates the following pattern.
- Use at least one *for* or *while* statement.

```
1 2 3 4 5 6 7
1 2 3 4 5 6
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

• Problem 4: Factorial function

- In the textbook, two different implementations of factorial function are introduced (Program 6-23, Program 6-24).
- Build a main function that gets a integer input from a user, calls both factorial implementations to calculate the factorial of the integer, and compares the results.
- The display result should be

```
Enter an integer: 3  
Result of 6-23: 6  
Result of 6-24: 6  
Results are the same.
```

• Problem 5: Geometric and harmonic mean

- Statisticians use many different algorithms in addition to the arithmetic average.
- Two other averages are the geometric and the harmonic mean.
- The geometric mean of a set of n numbers, $x_1, x_2, x_3, \dots, x_{n-1}, x_n$ is defined by the following formula:

$$\sqrt[n]{x_1 \times x_2 \times \dots \times x_n}$$

- The harmonic mean is defined by the following formula:

$$\frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \dots + \frac{1}{x_n}}$$

- Your program first gets several integer inputs from a user.
- The input process lasts until the user enter 99999.
- Once 99999 is entered, calculate the geometric and harmonic means of the numbers, and display the results as follows.

```
Enter integer: 7
Enter integer: 36
Enter integer: 1004
Enter integer: 99999
The geometric mean is 63.2477
The harmonic mean is 17.4794
```

• Problem 6: Prime number checking

- Write a prime-number checking program.
- The program first generates three random numbers (between 1 and 100) and checks whether each number is a prime number or not. Refer to the program code in the lecture note for prime number checking.
- The program should print for example

```
Three random numbers are generated. 10, 52, 3
10 is not a prime number.
52 is not a prime number.
3 is a prime number.
```

• Problem 7: Calculator with menu

- Extend your calculator program so that a user can now iteratively choose among four basic operations (+, -, *, /).
- The program should follow the following procedure
 1. First prompt a user to enter two integer numbers.
 2. Prompt the user to choose among the operations.
 1. If the input is '!', then end the program.
 2. Otherwise, conduct the operation and print the result.
 3. Go back to the step 2.
- Execution example

```
Enter two integers: 2 4
Choose operation: +
Result: 5
Choose operation: -
Result: -2
Choose operation: /
Result: 0.5
Choose operation: *
Result: 8
Choose operation: !
Bye~
```