



Programming Fundamentals

Programming Day - Week 07

Introduction

Welcome to your favorite day of the week which is programming day???. This week, we shall work together to solve complex problems using functions, conditional statements, loops and nested loops.

Let's do some coding.

Nested Loops

We can use loops inside other loops to perform even more complex and difficult tasks. Consider the following task for better understanding

Task 01(WP):

Write a program that inputs a number of rows from the user and prints that many number of asterisk rows on the console screen.

Notice, that there is a pattern in the output

- The first line has a single asterisk
- The second line has two asterisk
- 3rd line have 03 asterisk.
- 4th line have 04 asterisk.
- 5th line have 05 asterisk.
- And so on.

What if we can use a loop that controls each line (Rows) and an inner loop that will print that many asterisks (Columns) on the screen?

Consider the following solution for better understanding.



Programming



Fundamentals

Programming Day - Week 07

```
1  #include <iostream>
2  using namespace std;
3
4  void printStars(int rowSize);
5
6  main()
7  {
8      int rowSize;
9      cout << "Enter desired number of rows: ";
10     cin >> rowSize;
11     printStars(rowSize);
12 }
13
14 void printStars(int rowSize)
15 {
16     for (int row = 1; row <= rowSize; row++)
17     {
18         for(int col = 1 ; col <= row; col++)
19         {
20             cout << "+";
21         }
22         cout << endl;
23     }
24 }
25 }
```

Consider the attached solution.

Great Work!

Now attempt the following question on your own.

Task 01(OP): (Diamond)

Write a program using Nested Loops that generates the following shape on the console screen.

Notice the attached output and observe the pattern of the output.

Task 02(OP): (Half Diamond)

Write a program using Nested Loops that generates the following shape on the console screen.



- The given parameter **num** will always be equal to or greater than 1.
- Remember to include the num as the higher bound of the sequence.
- A number amplified by a factor b can also be read as $a * b$.

- A number **a** is exactly divisible by a number **b** when the remainder of the division a / b is equal to 0.



Remember the function is of **void** Type.



Programming Fundamentals

Programming Day - Week 07

Tests Cases

amplify(4) → 1, 2, 3, 40

// Create a sequence from 1 to 4

// 4 is exactly divisible by 4, so it will be $4 * 10 = 40$

amplify(3) → 1, 2, 3

// Create a sequence from 1 to 3

// There are no numbers that can be exactly divided by 4

amplify(25) → 1, 2, 3, 40, 5, 6, 7, 80, 9, 10, 11, 120, 13, 14, 15, 160, 17, 18, 19, 200, 21, 22, 23, 240, 25

// Create a sequence from 1 to 25

// The numbers exactly divisible by 4 are: 4 ($4 * 10 = 40$), 8 ($8 * 10 = 80$)... and so on.

```
G:\Semesters\Programming Fundamentals (Fall 2023)\Week 7\PD Tasks>Task4.exe
Enter the number to Amplify: 4
1, 2, 3, 40

G:\Semesters\Programming Fundamentals (Fall 2023)\Week 7\PD Tasks>Task4.exe
Enter the number to Amplify: 25
1, 2, 3, 40, 5, 6, 7, 80, 9, 10, 11, 120, 13, 14, 15, 160, 17, 18, 19, 200, 21, 22, 23, 240, 25
```

Task 04(CP): (Triangular Number)

This Triangular Number Sequence is generated from a pattern of dots that form a triangle.

The first 5 numbers of the sequence, or dots, are

1, 3, 6, 10, 15, 21 ...

This means that the first triangle has just one dot, the second one has three dots, the third one has six dots, and so on as shown in the figure.

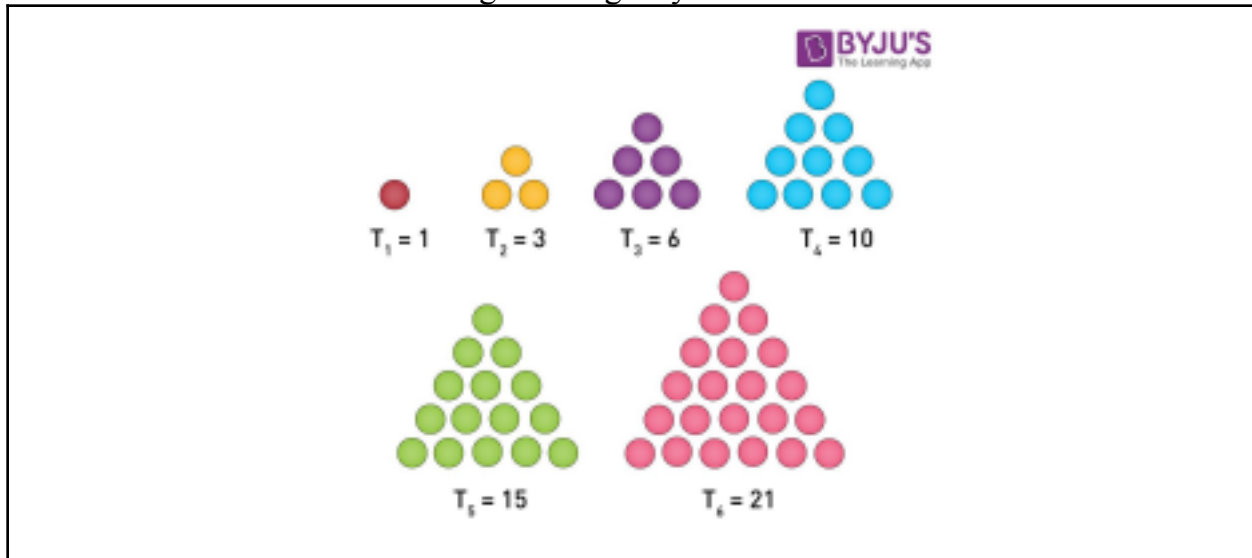


Programming



Fundamentals

Programming Day - Week 07



Write a function that returns the number of dots when given its corresponding triangle number of the sequence.

Test Cases:

`triangle(1) → 1`

`triangle(6) → 21`

`triangle(215) → 23220`

```
G:\Semesters\Programming Fundamentals (Fall 2023)\Week 7\PD Tasks>Task4.exe
Enter number of Triangle: 4
Dots in the Triangle: 10
G:\Semesters\Programming Fundamentals (Fall 2023)\Week 7\PD Tasks>Task4.exe
Enter number of Triangle: 15
Dots in the Triangle: 120
```

Task 05(CP): (isPrime)

Write a function that takes a number as input and returns whether the number is a prime number or not.

Prime numbers are those who are only divisible by 1 and its own.

Some examples of prime numbers are 2, 3, 5, 7, 11, 13, 17 etc.

`bool isPrime(int number); // function header`

Test Cases:

`isPrime(17) → 1`

`isPrime(15) → 0`



Programming Fundamentals



Programming Day - Week 07



Task 06(CP): (Primorial)

A *Primorial* is a product of the first n prime numbers (e.g. $2 \times 3 \times 5 = 30$). 2, 3, 5, 7, 11, 13 are prime numbers. If n was 3, you'd multiply $2 \times 3 \times 5 = 30$ or $\text{Primorial} = 30$.

Create a function that returns the Primorial of a number.

Test Cases:

`primorial(1)` → 2

`primorial(2)` → 6

`primorial(8)` → 9699690



Task 07(CP): (Hospital)

For a certain period of time, patients arrive at the hospital every day for an examination. It initially has 7 doctors. Each doctor can treat only one patient per day, but sometimes there is a shortage of doctors, so the remaining patients are sent to other hospitals. Every third day the hospital makes calculations and if the count of untreated patients is greater than the count of treated ones, another doctor is appointed. The appointment takes place before the daily patient acceptance begins.

Write a program that calculates for a given period of time, the count of treated and untreated patients.

Input Data

Input is read from the console and contains:



Programming Fundamentals

Programming Day - Week 07



On the first line – the period, for which you need to make calculations. Integer in the range of [1 ... 1000].

On the next lines (equal to the count of days) – count of the patients, who arrive for treatment for the current day. Integer in the range of [0 ... 10 000].

Output Data

Print on the console 2 lines:

On the first line: “Treated patients: {count of treated patients}.”

On the second line: “Untreated patients: {count of untreated patients}.”

Test Cases:

Input	Output Explanation
4 7 27 9 1	Treated patients: 23 Untreated patients: 21 Day 1: 7 treated and 0 untreated patients for the day Day 2: 7 treated and 20 untreated patients for the day Day 3: By this moment the treated patients are 14, and untreated ones – 20 → New doctor is appointed. → 8 treated and 1 untreated patients for the day Day 4: 1 treated and 0 untreated patients for the day Total: 23 treated and 21 untreated patients.
6 25 25 25 25 25 2 3 7 7	Treated patients: 40 Untreated patients: 87 Treated patients: 21 Untreated patients: 0



Programming Fundamentals



Programming Day - Week 07



Task 08(CP): (Cargo)

You are responsible for the logistics of various types of cargo. Depending on the weight of each cargo, you need a different vehicle, and this will cost a different price per ton: • Up to 3 tons – a minibus (200 USD per ton).

- From over 3 and up to 11 tons – truck (175 USD per ton).
- Over 11 tons – train (120 USD per ton).

Your task is to calculate the average price per ton of the cargo, and also what percentage of the cargo is transported in each vehicle.

Input Data:

From the console, we must read a sequence of numbers, each on a separate line: • First line: count of cargo for transportation – integer in the range of [1 ... 1000]. • On the next lines we pass the tonnage of the current cargo – integer in the range of [1 ... 1000].

Output Data:

Print on the console 4 lines, as follows:

- Line #1 – the average price per ton of the cargo (rounded up to the second digit after the decimal point).

- Line #2 – percentage of the cargo, carried by minibus (between 0.00% and 100.00%, rounded up to the second digit after the decimal point).
- Line #3 – percentage of the cargo, carried by truck (between 0.00% and 100.00%).
- Line #4 – percentage of the cargo, carried by train (between 0.00% and 100.00%).



Programming Fundamentals

Programming Day - Week 07

Test Cases:

Input	Output	Explanation
4 1 5 16 3	143.80 16.00% 20.00% 64.00%	By minibus you transport two of the cargo 1 + 3, total of 4 tons. By truck you transport one of the cargo: 5 tons. By train you transport one of the cargo: 16 tons. Sum of all cargo is: $1 + 5 + 16 + 3 = 25$ tons. Percentage of the cargo by minibus: $4/25 * 100 = 16.00\%$ Percentage of the cargo by truck: $5/25 * 100 = 20.00\%$ Percentage of the cargo by train: $16/25 * 100 = 64.00\%$ Average price per ton of carried cargo: $(4 * 200 + 5 * 175 + 16 * 120) / 25 = 143.80$

```

5      149.38
2      7.50%
10     42.50%
20     50.00%
1
7

4      120.35
53     0.00%
7      0.63%
56     99.37%
999

```

Hint: Use the following 2 lines to print output up-to 2 decimal points.

```
#include<iomanip>
```

```
cout << fixed;  
cout << setprecision(2);
```



Programming Fundamentals



Programming Day - Week 07



Good Luck and Best Wishes !!

Happy Coding ahead :)