

Lab Exercise - 1

Lab Exercise 1 — Random Data Generation, Mean, Variance and Standard Deviation Calculation

This problem is intended to be solved in a closed-lab session with a teaching assistant or instructor present. The problem is divided into five parts:

1. Lab Objectives
2. Description of the Problem
3. Sample Output
4. Program Template
5. Problem-Solving Tips

The program template represents a complete working C++ program. Read the problem description and examine the sample output; then study the template code. Using the problem-solving tips as a guide, implement the C++ code. Compile and execute the program. Compare your output with the sample output provided.

Lab Objectives

In this lab, you will practice:

- Using cout to output text and variables.
- Using cin to input data from the user.
- Using if statements to make decisions based on the truth or falsity of a condition.
- Using the arithmetic operators to perform calculations.
- Using relational operators to compare values.
- Using arrays to hold data
- Using loops to calculate some operations
- Using C++ references to reference a variable
- Using I/O manipulation to visualize output in a proper way

Description of the Problem

Write a C++ console application program that calculates the “Mean”, “Variance” and “Standard Deviation” of a randomly generated data. Program template and required function prototypes are given to you. Your job is to implement the given functions and visualize the generated data and calculated values on the command prompt. (Note: You are not allowed to change the given function prototypes or add a new function, just implement given functions)

Sample Output

```
+-----+
| USER INPUT |
+-----+
Enter the 'Data Size' (<0 for terminate program>) : 5
Enter the 'Minimum Number' : 4
Enter the 'Maximum Number' : 17
+-----+
| DATA ARRAY |
+-----+
5
11
9
15
11
+-----+
| MEAN |
+-----+
10.2
+-----+
| VARIANCE |
+-----+
10.56
+-----+
| STANDART DEVIATION |
+-----+
3.2496
+-----+
| EXECUTION FINISHED SUCCESSFULLY |
+-----+
Press any key to continue . . .
```

```
+-----+
| USER INPUT |
+-----+
Enter the 'Data Size' (<0 for terminate program>) : 0
+-----+
| TERMINATED BY USER! |
+-----+
Press any key to continue . . .
```

Template

```
// Include iostream library for input output operations
#include <iostream>
// Include time.h library for rand operations
#include <time.h>
// Include iomanip library for I/O manipulation operations
#include <iomanip>
// Include string library for string operations
#include <string>
using namespace std;
/// <summary>
/// Takes the user input.
/// </summary>
/// <param name="data_size">The data_size. (output)</param>
/// <param name="min_number">The min_number. (output)</param>
/// <param name="max_number">The max_number. (output)</param>
/// <returns>[true] if success, [false] otherwise</returns>
```

```
bool TakeUserInput(int& data_size, int& min_number, int& max_number);
/// <summary>
/// Creates the data set.
/// </summary>
/// <param name="dataSize">Size of the data.</param>
/// <param name="min_number">The min_number.</param>
/// <param name="max_number">The max_number.</param>
/// <returns>Pointer to an integer array that hold the random generated data</returns>
int* CreateDataSet(int dataSize, int min_number, int max_number);
/// <summary>
/// Calculates the mean.
/// </summary>
/// <param name="data_array">The data_array.</param>
/// <param name="data_size">The data_size.</param>
/// <returns>Mean value</returns>
double CalculateMean(int* data_array, int data_size);
/// <summary>
/// Calculates the variance.
/// </summary>
/// <param name="data_array">The data_array.</param>
/// <param name="data_size">The data_size.</param>
/// <param name="mean">The mean.</param>
/// <returns>Variance</returns>
double CalculateVariance(int* data_array, int data_size, double mean);

/// <summary>
/// Calculates the standart deviation.
/// </summary>
/// <param name="data_array">The data_array.</param>
/// <param name="data_size">The data_size.</param>
/// <param name="mean">The mean.</param>
/// <returns>Standart Deviation</returns>
double CalculateStandartDeviation(int* data_array, int data_size, double mean);
/// <summary>
/// Prints the message.
/// </summary>
/// <param name="message">The message.</param>
void PrintMessage(const string& message);
/// <summary>
/// Prints the data array.
/// </summary>
/// <param name="data_array">The data_array.</param>
/// <param name="data_size">The data_size.</param>
void PrintdataArray(int* data_array, int data_size);
/// <summary>
/// Prints the mean.
/// </summary>
/// <param name="mean">The mean.</param>
void PrintMean(double mean);
/// <summary>
/// Prints the variance.
/// </summary>
/// <param name="variance">The variance.</param>
void PrintVariance(double variance);
/// <summary>
/// Prints the standart deviation.
/// </summary>
/// <param name="std_deviation">The std_deviation.</param>
void PrintStandartDeviation(double std_deviation);

/// <summary>
/// Main function.
/// </summary>
```

```

/// <returns></returns>
int main()
{
    srand(time(NULL));
    int dataSize, minNumber, maxNumber;
    if (!TakeUserInput(dataSize, minNumber, maxNumber)){
        PrintMessage("TERMINATED BY USER!");
        return 1;
    }

    int* DataArray = CreateDataSet(dataSize, minNumber, maxNumber);
    PrintdataArray(DataArray, dataSize);
    double mean = CalculateMean(DataArray, dataSize);
    PrintMean(mean);
    double variance = CalculateVariance(DataArray, dataSize, mean);
    PrintVariance(variance);
    double stdDeviation = CalculateStandartDeviation(DataArray, dataSize, mean);
    PrintStandartDeviation(stdDeviation);
    PrintMessage("EXECUTION FINISHED SUCCESSFULLY");
    return 0;
}

```

Problem-Solving Tips

1. “TakeUserInput” function receives three reference arguments and changes this variable values and returns true or false depending on user input.
2. To create a randomly generated dataset you can use “rand()” function. However, you have to generate random numbers depending on the minimum and maximum values. That is, generated numbers must be in the given range. To generate a random number between the given range. You can use rand() function as given:

```
int randomNumber = min_number + rand() % (max_number - min_number + 1);
```

3. To calculate the “Mean” of a data set, look at the given mean calculation formula.

$$\mu = \frac{\sum_{i=1}^n x_i}{n}$$

μ : Mean

x_i : A member of a dataset

n : Number of elements in the dataset

4. To calculate the “Variance” of a data set, look at the given variance calculation formula.

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)^2$$

σ^2 : Variance of the dataset

x_i : A member of a dataset

n : Number of element in the dataset

μ : Mean

5. To calculate the “Standart Deviation” of a data set, look at the given standart deviation calculation formula.

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{n}}$$

σ : Standart Deviation

x_i : A member of a dataset

n : Number of element in the dataset

μ : Mean