Computer Programming

Dr. Deepak B Phatak
Dr. Supratik Chakraborty
Department of Computer Science and Engineering
IIT Bombay

Session: Using Arrays for solving computational problems

Quick Recap

IIT Bombay

- C++ provides array data structure
- An array
  - is declared with a type and a size
  - elements of an array are accessed by using index expressions
  - value of index must be between 0 and size-1

Quick Recap

IIT Bombay

- Iterative constructs can be easily used to examine and process all elements of an array, in some desired order
- A for loop is ideal to iterate on values of index i, from 0 to N-1

```
for (i=0; i < N; i++){
    //  code to process array elements
}
```

Overview of This Lecture

IIT Bombay

- Using arrays to solve computational problems
  - Finding average and standard deviation of marks scored in quiz

Problem: Calculating Marks Statistics

N students of a class have taken a quiz. Their marks are to be processed to calculate the average, and standard deviation. Assume that there are at most 600 students in the class.

Analysis:

- We have seen how to solve the problem of finding the average of given numbers. We will use the same algorithm
  - Declare an array MARKS of size 600, type integer (?)
  - Read and store N marks in an array MARKS
  - Find the sum of all N marks, divide by N to get average (μ)

Standard Deviation

IIT Bombay

- We need to know how to calculate standard deviation
- Standard deviation is a statistical term. Its value describes how much various data values are spread around average value.
  - A large value indicates many data values are far away from average
  - A smaller value indicates most data values to be close to average.
- Useful in many practical situations

Standard Deviation

IIT Bombay

If μ is the average of N marks, standard Deviation (σ)is defined in terms of our array elements, as:

$$\sigma = \text{sqrt}\left(\sum 1/N * (MARKS[i] - \mu)^2\right),$$

(where the summation is over i = 0 to N-1)

sqrt represents the square root of the quantity in parenthesis

- C++ provides a library, with many mathematical functions
- We must include a header file called <cmath> in our program, in order to use the sqrt, and other functions

Program Design

IIT Bombay

- We use familiar algorithm to calculate average
  - Read and store N marks in an array MARKS
  - Iteratively find the sum, divide by N to get average mu (μ)
- Next we calculate standard deviation sigma (σ)
  - Set up another iteration over i, to scan all array elements again
    - Calculate the difference diff, between marks[i] and μ
    - Add the square of diff to the running sum
  - Divide the sum by N, and calculate square root, to get sigma

Program to find statistics of marks scored

IIT Bombay

```
int main(){
   // program to find marks statistics
   int marks[600] , i, N;
   float sum, diff, mu, sigma;
   // read N elements of marks array
   cin >> N;
   for (i =0; i < N; i = i +1){
        cin >> marks[i];
   }
```

Program … (average)

IIT Bombay

```
// Calculate average mu
sum = 0.0;
for (i =0; i < N; i = i +1){
        sum = sum + marks[i];
}
mu = sum/N;
```

Program … (standard deviation)

IIT Bombay

```
// Calculate standard deviation sigma
sum = 0.0;
for (i =0; i < N; i = i +1){
        diff = marks[i] - mu
        sum = sum + diff * diff;
}
sigma = sqrt (sum/N);
```

Program … (output results)

IIT Bombay

```
   cout << “Statistics of quiz 1 marks of the class” << endl;
   cout << “Number of students is: “ << N << endl;
   cout  << “Average marks: “ << mu << endl;
   cout  << “Standard Deviation of Marks: “ << sigma << endl;
return 0;
}
```

A few practical aspects of programming

- Users may commit mistakes while giving data
- What if the value of N is given as 1000?
  - How do we convey to users, that the limit is 600 marks?
- At least, our program should validate input
  - Check if value of N is out of bound
  - If so, give an error message, and exit with a value other than 0

Practical programming …

IIT Bombay

```
// validate input
if (N < 1 || N > 600){
    cout << “Invalid value of N: “ << N << endl;
    return -1;
}
```

Practical Programming

- For solving any problem, it is useful to
  - Write a small design document , outlining the algorithm
  - Indicate variables and arrays to be used
- While writing a program
  - Write program segments in logical order
  - Write explanatory comments
  - Use simple, and easy to understand constructs

Summary

- We have learnt how to use arrays to solve simple computational problems
- In the next session, we will consider another problem