SOMAIYA
VIDYAVIHAR UNIVERSITY
K J Somaiya School of Engineering
(formerly K J Somaiya College of Engineering)

Structured Programming Methodology

By

Prof. Vaibhav P. Vasani

Assistant Professor

Department of Computer Engineering

K. J. Somaiya School of Engineering

Somaiya Vidyavihar University

vaibhav.vasani@somaiya.edu,
vaibhav.vasani@gmail.com

Somaiya
TRUST

- A flag variable is simply a variable (usually of type bool or int) used to indicate the presence, absence, or status of a condition during program execution.

Use of flags

- To keep track of whether something has happened.
- To control the flow of loops and decisions.
- To avoid repeated checks or computations.

Example 1: Flag with Loop (Searching in Array)

```
#include <iostream>
using namespace std;

int main() {
    int arr[5] = {10, 25, 30, 45, 60};
    int key, flag = 0;

    cout << "Enter a number to search: ";
    cin >> key;

    for (int i = 0; i < 5; i++) {
        if (arr[i] == key) {
            flag = 1;   // condition met → set flag
            break;
        }
    }

    if (flag == 1)
        cout << "Number found in the array!" << endl;
    else
        cout << "Number not found!" << endl;

    return 0;
}
```

Example 2: Boolean Flag

```
#include <iostream>
using namespace std;

int main() {
    bool isPrime = true;
    int n;

    cout << "Enter a number: ";
    cin >> n;

    if (n <= 1)
        isPrime = false;
    else {
        for (int i = 2; i <= n / 2; i++) {
            if (n % i == 0) {
                isPrime = false;   // flag set
                break;
            }
        }
    }

    if (isPrime)
        cout << n << " is Prime." << endl;
    else
        cout << n << " is Not Prime." << endl;

    return 0;
}
```

Key characteristics and uses of flags in C++

- **Signaling Conditions:**
  - Flags are primarily used to indicate whether a certain condition has been met or a specific event has occurred. For example, a found_item flag could be set to true once an item is located in a search operation.
- **Controlling Program Flow:**
  - The value of a flag can be used in conditional statements (like if-else or switch) to determine which code path to execute. This allows for dynamic behavior based on runtime conditions.
- **Synchronization in Concurrency:**
  - In multithreaded programming, flags can be used as simple synchronization mechanisms. For instance, a flag can signal that a shared resource is currently in use, preventing other threads from accessing it simultaneously.
- **Indicating Options/Settings:**
  - Flags can represent program settings or options, where different values correspond to different configurations. This is common in command-line arguments, where flags like -v (verbose) or -d (debug) can alter program output or behavior.
- **Error Handling:**
  - A flag can be set to indicate the presence of an error, allowing subsequent code to handle the error condition gracefully.

SOMAIYA VIDYAVIHAR UNIVERSITY
K J Somaiya School of Engineering
(formerly K J Somaiya College of Engineering)

Key Points about Flags

- **Default value** is often 0 (false) or false.
- Flags are **changed when a condition occurs**.
- Useful for **breaking loops**, **validating input**, or **status checking**.

Somaiya TRUST

SOMAIYA
VIDYAVIHAR UNIVERSITY
K J Somaiya School of Engineering
(formerly K J Somaiya College of Engineering)

Practice Questions on Flag

SOMAIYA
VIDYAVIHAR UNIVERSITY
K J Somaiya School of Engineering
(formerly K J Somaiya College of Engineering)

SOMAIYA
VIDYAVIHAR UNIVERSITY
K J Somaiya School of Engineering
(formerly K J Somaiya College of Engineering)

SOMAIYA
VIDYAVIHAR UNIVERSITY
K J Somaiya School of Engineering
(formerly K J Somaiya College of Engineering)