

# Variabel Array dan String

Pertemuan 4

# Outline

- Arrays and Strings
  - Storing in Arrays
  - Multidimensional Arrays
  - Character Arrays
  - Copying Strings

# Definisi Array

- Array adalah kumpulan data yang saling terkait dan memiliki tipe data yang sama. Array dapat dibayangkan sebagai serangkaian lokasi penyimpanan data. Setiap lokasi penyimpanan disebut elemen array.
- Array dideklarasikan dinyatakan dengan menulis tipe data dan nama array diikuti dengan jumlah elemen array memegang dalam kurung siku. Berikut ini contohnya:

```
long peaks[25];           // initialization
peaks[0] = 29029;        // first element
peaks[24] = 7804;       // last element
```

# Inisialisasi Array

- Inisialisasi array dilakukan dengan dengan menuliskan banyaknya diikuti isi dari array. Jika array terlalu banyak, isi index dari array dapat dapat dikosongkan, compiler akan secara otomatis menghitung jumlah elemen dari array. Fungsi `sizeof()` dapat digunakan untuk menghitung jumlah elemen dari array.

```
int post[5] = {10,20,30,40,50};  
int post[] = {0,10,20,30,40,50,60,70,80,90}  
const int size = sizeof(post)/sizeof(post[0]);
```

- Berikut ini cara penulisan array multi dimensi

```
int grid[5,13];  
int cube[5,13,8];
```

# Strings : Character Arrays

- String adalah array dari karakter yang diakhiri dengan karakter null atau '\0'. Anda dapat mendeklarasikan dan menginisialisasi string seperti array lainnya.
- Karena karakter-karakter sulit untuk mengetik dan mengantisipasi kesalahan, C++ memungkinkan menyingkat dari inisialisasi string menggunakan literal:

```
char hello[]={ 'H', 'e', 'l', 'l', 'o', ' ', 'w', 'o', 'r', 'l', 'd', '\0' };  
char hello[]="Hello world";
```

- Terdapat library fungsi untuk menangani string yang tergabung dalam program dengan termasuk string.h dengan menambahkan "#include <string.h>" ada banyak fungsi yang disediakan dua untuk menyalin satu string menjadi lain strcpy () dan strncpy ().

# 1. Storing in Arrays

## WeightGoals.cpp

```
1: #include <iostream>
2:
3: int main()
4: {
5:     float goal[4];
6:     goal[0] = 0.9;
7:     goal[1] = 0.75;
8:     goal[2] = 0.5;
9:     goal[3] = 0.25;
10:    float weight, target;
11:
12:    std::cout << "Enter current weight: ";
13:    std::cin >> weight;
14:    std::cout << "\nEnter goal weight: ";
15:    std::cin >> target;
16:    std::cout << "\n";
17:
18:    for (int i = 0; i < 4; i++)
19:    {
20:        float loss = (weight - target) * goal[i];
21:        std::cout << "Goal " << i << ": ";
22:        std::cout << weight - loss << "\n";
23:    }
24:
25:    return 0;
26: }
```

# 2. Multidimensional Arrays

## Box.cpp

```
1: #include <iostream>
2:
3: int main()
4: {
5:     int box[5][3] = { 8, 6, 7, 5, 3, 0, 9, 2, 1, 7, 8,
6:                     9, 0, 5, 2 };
7:     for (int i = 0; i < 5; i++)
8:     {
9:         for (int j = 0; j < 3; j++)
10:        {
11:            std::cout << "box[" << i << "]";
12:            std::cout << "[" << j << "] = ";
13:            std::cout << box[i][j] << "\n";
14:        }
15:    }
16: }
```

# 3. Character Arrays

## BridgeKeeper.cpp

```
1: #include <iostream>
2:
3: int main()
4: {
5:     char name[50];
6:     char quest[80];
7:     char velocity[80];
8:
9:     std::cout << "\nWhat is your name? ";
10:    std::cin.getline(name, 49);
11:
12:    std::cout << "\nWhat is your quest? ";
13:    std::cin.getline(quest, 79);
14:
15:    std::cout << "\nWhat is the velocity of an unladen swallow? ";
16:    std::cin.getline(velocity, 79);
17:
18:    std::cout << "\nName: " << name << "\n";
19:    std::cout << "Quest: " << quest << "\n";
20:    std::cout << "Velocity: " << velocity << "\n";
21:    return 0;
22: }
```



# 4. Copying Strings

## StringCopier.cpp

```
1: #include <iostream>
2: #include <string.h>
3:
4: int main()
5: {
6:     char string1[] = "Free the bound periodicals!";
7:     char string2[80];
8:
9:     strcpy(string2, string1);
10:
11:     std::cout << "String1: " << string1 << std::endl;
12:     std::cout << "String2: " << string2 << std::endl;
13:     return 0;
14: }
```

# Tugas

- Buatlah sebuah program yang menanyakan kepada user, nama depan dan nama belakang, kemudian tampilkan bersamaan dengan salam untuk menyapa user tersebut.
- Modifikasi program `WeightGoals.cpp` dan tambahkan 2 intermediate milestones baru untuk 90% dan 80%.