PEMROGRAMAN DASAR KELAS XI SEMESTER II

ARRAY

- Mengapa kita harus menggunakan array?
  - *Misal kita akan membuat program yang menyimpan nilai pemdas untuk 5 siswa. Maka sewajarnya kita menuliskan :*
    - int siswa1 = 70;
    - int siswa2 = 80;
    - int siswa3 = 90; dan seterusnya

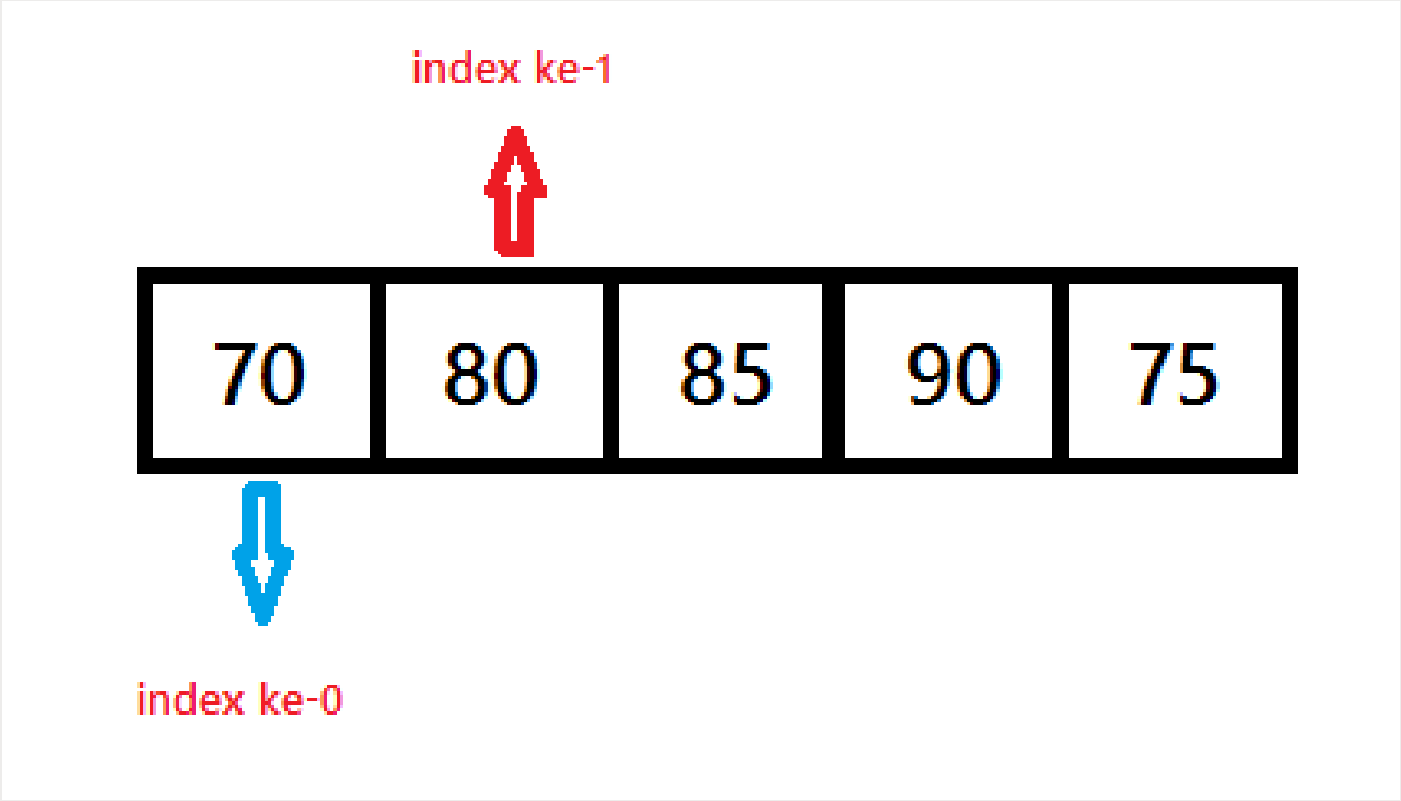
    Nah, bagaimana kalau data siswa ada 1000? Mungkinkah kita mendeklarasikan

    int siswa1 hingga int siswa1000?

    Jawabannya adalah mungkin, namun TIDAK EFISIEN sehingga akan memakan memori lebih banyak di komputer.

apabila data tergolong SEJENIS dan memiliki TIPE DATA yang sama, gunakan bantuan ARRAY agar LEBIH EFISIEN

- Kasus sebelumnya, apabila kita akan menyimpan nilai pemdas untuk 5 siswa, kita cukup mendeklarasikannya seperti di bawah ini:
  - *STRUKTUR UMUM*
    - (tipedata) (namavariabel) [jumlahruang] = {isi setiap ruang}
  - *PENERAPAN*
    - int nilaipemdas [5] = {70,80,85,90,75};
- Yang harus diingat adalah, ARRAY dimulai dari INDEX KE-0 yang berarti 70 menempati ruangan ke-0



cara MEMANGGIL ISI array adalah dengan MEMANGGIL INDEX-nya

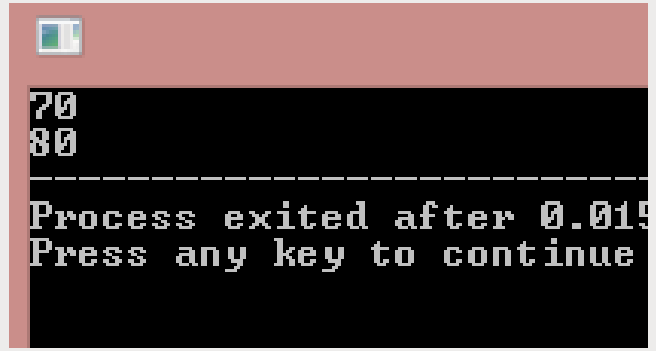
PROGRAM UTUH

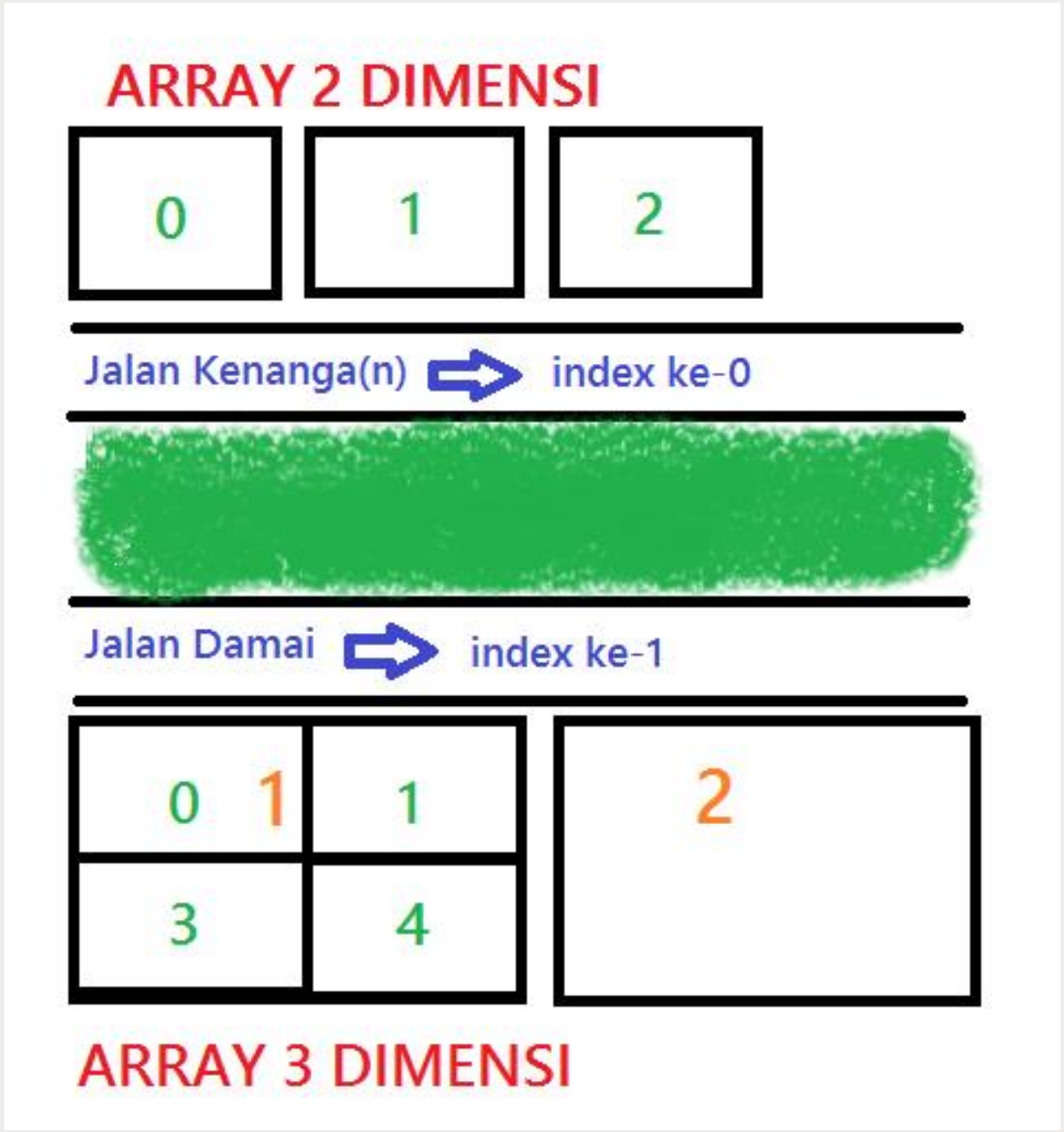
```cpp
#include <iostream>
using namespace std;

int main() {
    //mendeklarasikan array bertipe data INT
    int nilaipemdas [5] = {70,80,85,90,75};

    //memanggil isi array pada ruangan pertama
    cout << nilaipemdas[0];
    //memanggil isi array pada ruangan kedua
    cout << nilaipemdas[1];
    //dan seterusnya
}
```

OUTPUT





ARRAY 2 DIMENSI

struktur umum :
(tipedata) (namavar) [jmlbaris] [jmlkolom] = {isi};

contoh :
di jalan kenanga(n), terdapat 3 buah rumah masing-masing beranggotakan 3,5,7 orang.

syntax :
int anggota[1][3] = {3,5,7};

penjelasan :
1 menunjukkan jumlah jalan (baris)
3 menunjukkan jumlah rumah (kolom)
3,5,7 menunjukkan anggota keluarga (isi)

cara memanggil :
//memanggil rumah beranggotakan 3
cout << anggota[0][0];
ingat bahwa array dimulai dari index ke-0
//memanggil rumah beranggotakan 5
cout << anggota[0][1];
//memanggil rumah beranggotakan 7
cout << anggota[0][2];

jika ada jalan lain, 1 dapat diubah sesuai jumlah jalan, begitu pula jumlah rumah, angka 3 bisa disesuaikan dengan jumlah rumah.

ARRAY 3 DIMENSI

struktur umum :
(tipedata) (namavar) [jmlbaris] [jmlkolom] [jmlkarakter] = {isi};

contoh :
di jalan Damai terdapat 2 rumah. Rumah pertama terdiri dari 4 ruangan yang masing-masing dihuni oleh 1,2,3, dan 4 anak kos.

syntax :
int anggota[1][2][4] = {1,2,3,4};

penjelasan :
1 menunjukkan jumlah jalan (baris) yaitu jalan Damai
2 menunjukkan jumlah rumah (kolom) di jalan Damai
1,2,3,4 menunjukkan anak kos (isi)

cara memanggil :
//memanggil kos beranggotakan 1
cout << anggota[0][0][0];
ingat bahwa array dimulai dari index ke-0
//memanggil rumah beranggotakan 2
cout << anggota[0][0][1];
//memanggil rumah beranggotakan 3
cout << anggota[0][0][2];
//memanggil rumah beranggotakan 4
cout << anggota[0][0][3];
jika ada jalan lain, 1 dapat diubah sesuai jumlah jalan, begitu pula jumlah rumah, angka 2 bisa disesuaikan dengan jumlah rumah, dan angka 4 bisa disesuaikan dengan jumlah ruangan.