**Nama :** Ifham Syafwan Fikri

**Github** **:** <https://github.com/FikriSyafwan/Penugasan-Praktikum-Pemrograman>

**Nomor Mahasiswa :** 24/545184/PA/23161

**Kelas :** KOM B

**Dosen Pengampu :** Muhammad Husni Santriaji

**LAPORAN PRAKTIKUM PEMROGRAMAN**

**PERTEMUAN 6**

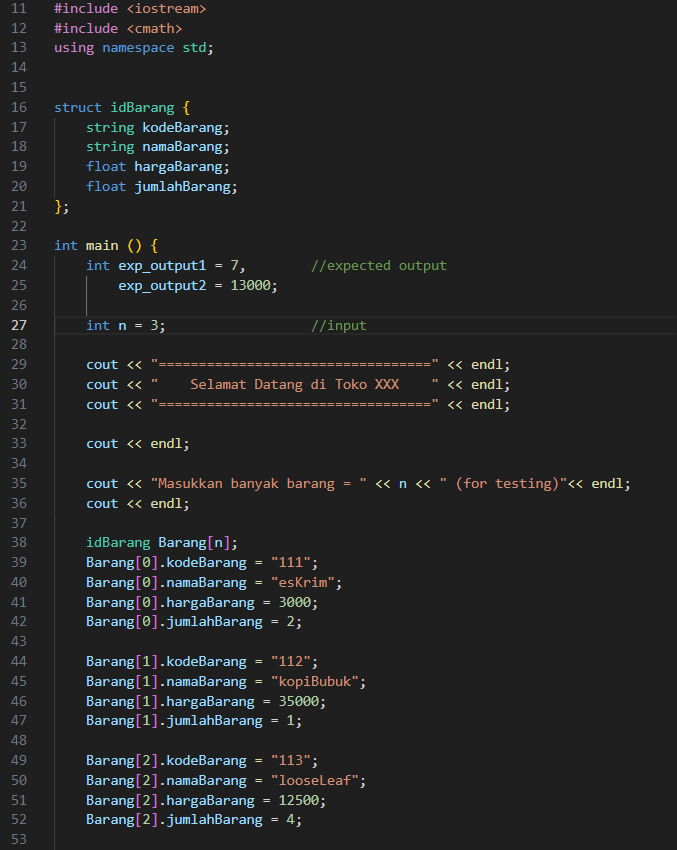
**Latihan**

**Number 1:** Implementasi Struct & Array pada pencatatan barang

Pada praktikum pemrograman pertemuan keenam, permasalahan yang diangkat berupa pencatatan barang swalayan. Pencatatan yang dimaksud adalah pendataan barang dengan beberapa *variable*, antara lain kode barang, nama barang, harga barang, dan jumlah barang serta perhitungan jumlah barang sebanyak “n” dan harga rata-rata dari “n” jumlah barang.

Pada permasalahan ini, struct dan array digunakan untuk mempermudah pengerjaan. Struct itu apa? Struct atau *structures* merupakan struktur yang berfungsi untuk menyimpan atau memuat berbagai macam tipe data dalam satu bagian. Setiap *variable* pada struct disebut *member of structure*. Berikut merupakan program yang menghasilkan *output* atau keluaran dari permasalahan di atas:

1. struct, expected output, dan input,

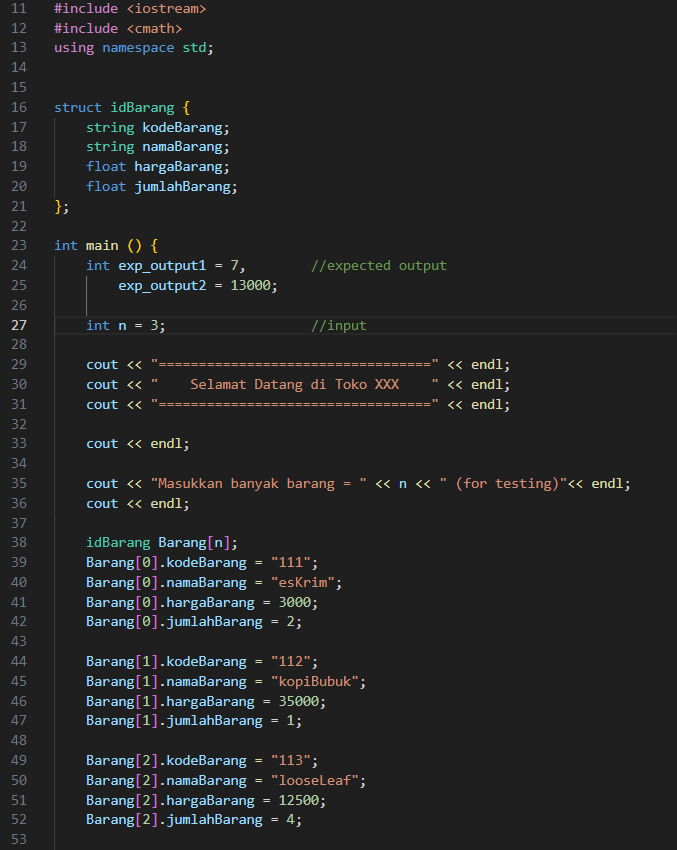


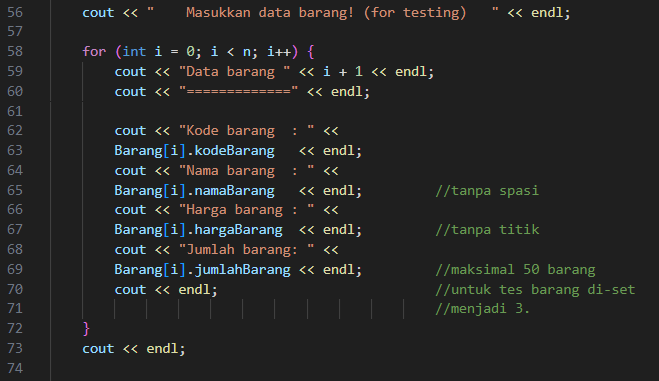
Struct akan diberikan nama “idBarang” dalam program ini dengan beberapa *member of structure* berupa

* string kodeBarang : kode unik untuk mempresentasikan suatu barang,
* string namaBarang : nama dari barang,
* float hargaBarang : harga barang, dan
* float jumlahBarang : jumlah barang.

Pada int main() akan diberikan *input* = 3 dan *expected output* = exp\_output1: 7 (jumlah barang) dan exp\_output2: 13.000 (harga rata-rata barang) untuk menguji apakah program akan berjalan lancar dan menghasilkan *output* atau keluaran yang sesuai dengan hipotesis. Apabila sesuai dengan expected output, program akan menghasilkan keluaran “[Test succeed!! :>]”, sedangkan sebaliknya program akan menghasilkan keluaran “[Test failed?! :<].”

1. input barang menggunakan array,



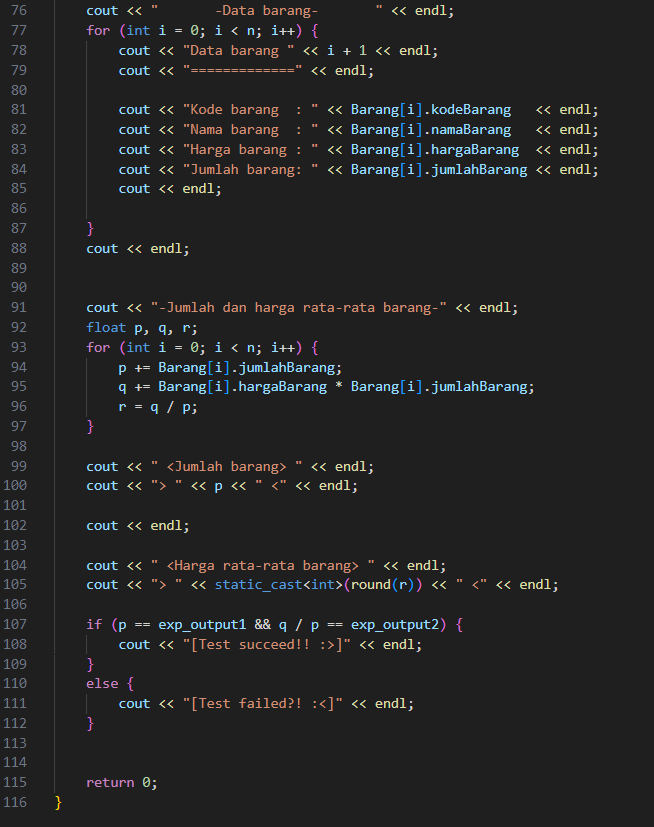


Berikut merupakan input dari tiga barang sebagai tes dengan menggunakan array pada program. Array itu apa? Berbeda dengan struct, array adalah kumpulan-kumpulan *variable* yang menyimpan data dengan tipe yang sama. Indeks dari elemen array dimulai dari angka 0, bukan angka 1. Dalam array, setiap *variable* dapat dibedakan dengan *variable* lain berdasarkan *subscript*. Bilangan dalam kurung siku […] disebut *subscript* dengan masing-masing elemen dapat diakses.

Array pada program akan dinamakan sebagai “Barang” dan dihubungkan dengan struct idBarang menjadikan setiap elemen pada Barang[i] memiliki *variable* dari struct idBarang. Untuk mengubah nilai dari masing-masing *variable*, dot syntax (.) diperlukan, sebagai contoh Barang[0]. kodeBarang pada program akan diberi string “111”.

Setelah memberikan nilai pada masing-masing *variable* untuk tiap elemen Barang[i], program akan mengeluarkan data dari input yang telah disimpan.

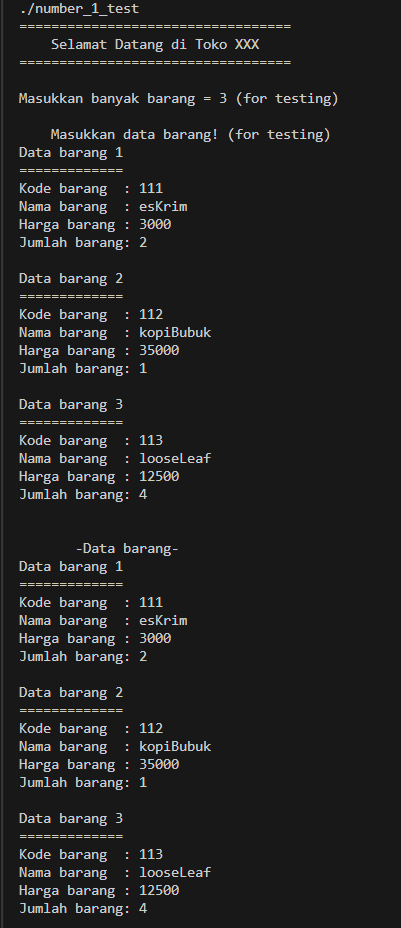
1. dan perhitungan jumlah dan harga rata-rata barang

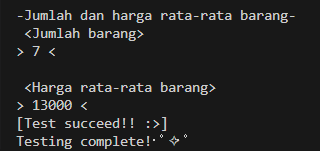


Untuk menghitung jumlah barang tiap elemen dari array Barang[i], tipe perulangan for akan digunakan sebanyak tiga kali, nilai i akan dimulai dari 0 karena indeks elemen dari array dimulai dari 0. Pada program ini juga, untuk memudahkan perhitungan, *variable* float p, q, dan r akan diberikan untuk menyimpan nilai. *Variable* p sebagai jumlah dari semua barang, q sebagai jumlah dari semua barang dikali harga barang, dan float r sebagai harga rata-rata dari barang yang telah di-*input* sebelumnya.

Kemudian, pembulatan akan diperlukan untuk menyamakan hasil dengan expected output. Pembulatan ditandai dengan static\_cast<int>(round(r)) menyebabkan #include <cmath> diperlukan.

Berikut merupakan hasil dari program yang telah diberi *input*:





**PERTANYAAN**

**Question 1:** Perbedaan tipe data *struct* dan *array*

***Array*** merupakan sebuah *variable* yang menyimpan lebih dari satu elemen dengan tipe data yang sama. *Array* dapat diakses dengan cara memanggil nama khusus dari *array* itu sendiri ditambah dengan notasi “[…]” yang berisi indeks. Indeks yang dimaksud adalah jumlah dari elemen yang ada dari suatu *array* dengan indeks angka 0 sebagai elemen pertama. Berikut merupakan contoh dari sintaks *array*:

int myArray[2];

myArray[2] = {1, 2};

***Struct*** merupakan sebuah *variable* yang menyimpan lebih dari satu elemen, sama seperti *array*. Namun, *array* dapat memuat lebih dari satu tipe data. Untuk memanggil *variable* dalam suatu struct*, variable struct* dapat diakses atau diberi suatu nilai dengan cara memanggil nama *struct* ditambah dengan notasi “.” yang kemudian ditambah dengan nama *variable* itu sendiri. Berbeda dengan *array*, *struct* tidak memiliki indeks untuk mendefinisikan nilai pada struktur. Berikut merupakan contoh dari sintaks *struct*:

struct myStruct {int kucing;};

myStruct.kucing = 1;

**Question 2:** Apakah tipe data struktur dapat berupa matriks dua dimensi atau lebih

Ya, kita bisa membuat matriks dua dimensi (atau lebih) menggunakan array.

1. **Mendefinisikan Struktur**: Struktur adalah tipe data khusus yang bisa menyimpan beberapa jenis data di dalam satu unit. Misalnya, kita bisa membuat struktur bernama Data yang memiliki anggota id, value, dan label, masing-masing mewakili tipe data yang berbeda (misalnya, int, float, dan char).
2. **Membuat Matriks dari Struktur**: Dengan menggunakan array dua dimensi, kita bisa membentuk matriks yang berisi elemen-elemen bertipe struktur Data. Misalnya, jika kita ingin membuat matriks berukuran 2x3, setiap elemen dalam matriks ini akan memiliki anggota id, value, dan label.
3. **Mengisi Data dalam Matriks**: Untuk mengubah data dalam matriks, kita bisa menggunakan indeks baris dan kolom.

Jadi, pada dasarnya, kita bisa membuat matriks dengan *struct* yang berisi elemen-elemen berstruktur menggunakan *array*.