

Running Modul Struktur Data CCH1D4 Latihan Pengenalan C++

/ CREATE PROJECT *****/**

1. Silahkan buka Code::Blocks pada komputer praktikum anda!
2. Buatlah sebuah project baru dengan perintah: **File >> New >> Project..**
3. Pilih **Console Application** pada jendela **New from template**, kemudian pilih **Next** hingga muncul pilihan bahasa yang digunakan.
4. Pilih **C++**, kemudian **Next**. Isikan field yang tersedia:

Project title:	RunMod_NIM	e.g. RunMod_1013710049
Folder to create project in:	D:\Running_Modul_Kelas\	e.g. D:\Running_Modul_IT-43-20\

5. Kemudian pilih **Next**, lalu **Finish**.

/ SOAL LATIHAN *****/**

Masih ingat soal nomor satu tentang **Himpunan** pada UAS mata kuliah Dasar Algoritma dan Pemrograman semester lalu? Apapun jawaban kalian, sekarang mari kita implementasikan jawaban soal tersebut ke dalam program dengan bahasa C++. Tidak perlu khawatir, ini bukan ujian, jadi jawabannya soal tersebut akan diberikan. Anda hanya diminta menulis ulang sintak pseudocode ke dalam bahasa C++ tentunya. Semangat ya 😊!

Himpunan adalah kumpulan obyek yang berbeda, dimana tidak ada duplikasi obyek dan obyek tidak harus dalam urutan tertentu. Diberikan dua buah himpunan, program dibawah ini akan memeriksa apakah kedua himpunan tersebut sama atau berbeda. Lengkapilah agar program dapat bekerja dengan benar.

Contoh interaksi (underlined berarti input)

Contoh 1:

Anggota himpunan 1: 8 2 4 1 5 4

Anggota himpunan 2: 8 2 4 1 5 1

Himpunan 1 = Himpunan 2? True

Contoh 3:

Anggota himpunan 1: 8 2 4 1 8

Anggota himpunan 2: 2 1 5 8 2

Himpunan 1 = Himpunan 2? False

Contoh 2:

Anggota himpunan 1: 8 2 4 1 5 4

Anggota himpunan 2: 2 1 5 8 4 1

Himpunan 1 = Himpunan 2? True

Contoh 4:

Anggota himpunan 1: 8 2 4 1 8

Anggota himpunan 2: 2 1 5 2

Himpunan 1 = Himpunan 2? False

/ TO DO LIST *****/**

Edit file main.cpp yang telah terbentuk ketika membuat project, sesuai dengan instruksi berikut ini: (**GUNAKAN MODUL PRAKTIKUM YANG DIBERIKAN**)

1. Tuliskan konstanta dan tipe data bentukan berikut ini

```
const nMax: integer = 37
type THimpunan <
    anggota : array [0..nMax-1] of integer
    panjang : integer
>
```

pada baris sebelum function main().

```
using namespace std;
// tuliskan pada bagian ini
...
...
...
int main();
```

2. Tuliskan subprogram berikut pada baris selanjutnya (masih sebelum function main). Penulisan function pada C++ harus diawali oleh return type kemudian baru nama function-nya. Contoh **bool** anggotaHimpunan(...)

```
function anggotaHimpunan(himp: THimpunan, oby:integer) -> boolean
/* mencari oby di dalam himpunan himp. True jika ada, False jika tidak
   ditemukan atau himpunan kosong - SEQUENTIAL SEARCH */
kamus
    found : boolean
    k : integer
algoritma
    found <- false
    k <- 0
    while (k <= himp.panjang and not found ) do
        found <- oby == himp.anggota[k]
        k <- k + 1
    endwhile
    return found
endfunction
```

Untuk **parameter output** gunakan simbol *ampersand* "&" sebelum nama variabel. Contohnya &himp.

Untuk perintah **input/read** gunakan **cin >>**, contohnya **input(N)** ditulis **cin >> N;**

```
procedure inputAnggotaHimpunan(output himp: THimpunan)
/* IS. data tersedia di dalam piranti masukan
   Proses: proses input akan berhenti apabila bilangan sudah pernah ada di
   dalam himpunan atau sudah mencapai kapasitas maksimum dari
   arraynya.
   FS. Terdefinisi himpunan bilangan bulat beserta panjangnya */
kamus
    bilangan : integer
algoritma
    himp.panjang <- -1
    input(bilangan)
    while (himp.panjang < nMax-1 and not anggotaHimpunan(himp,bilangan))do
        himp.panjang <- himp.panjang + 1
        himp.anggota[himp.panjang] <- bilangan
        input(bilangan)
    endwhile
endprocedure
```

Untuk **parameter input/output** gunakan simbol ampersand "&" seperti parameter output.

```
procedure urutAnggotaHimpunan(input/output himp: THimpunan)
/* IS. himpunan sudah terdefinisi
   Proses: urutkan anggota himpunan secara ascending dengan SELECTION SORT
   FS. Anggota himpunan sudat terurut */
kamus
    pass, i, idx_min, temp : integer
algoritma
    pass <- 1
    while (pass <= himp.panjang - 1) do
        idx_min <- pass - 1
        i <- pass
        while (i <= himp.panjang - 1) do
            if himp.anggota[idx_min] > himp.anggota[i] then
                idx_min <- i
            endif
            i <- i + 1
        endwhile
        temp <- himp.anggota[pass-1]
        himp.anggota[pass-1] <- himp.anggota[idx_min]
        himp.anggota[idx_min] <- temp
        pass <- pass + 1
    endwhile
endprocedure
```

penulisan parameter yang memiliki tipe data sama harus ditulis terpisah.

```
function himpunanSama(himp1,himp2 : THimpunan) -> boolean
/* mengembalikan TRUE jika himpunan 1 dan himpunan 2 memiliki anggota yang
   sama, dan FALSE jika sebaliknya */
kamus
    similar : boolean
    k : integer
algoritma
    urutAnggotaHimpunan(himp1)
    urutAnggotaHimpunan(himp2)

    similar <- himp1.panjang == himp2.panjang
    k <- 0
    while (k < himp1.panjang and similar) do
        similar <- himp1.anggota[k] == himp2.anggota[k]
        k <- k + 1
    endwhile
    return similar
endfunction
```

3. Tuliskan program utama berikut ini pada **function** main()

```
program HimpunanBilanganBulat
kamus
    set1, set2 : THimpunan
algoritma
    output("Anggota himpunan 1: ")
    /** Panggil subprogram untuk membaca anggota himpunan 1 */
    inputAnggotaHimpunan(set1)

    output("Anggota himpunan 2: ")
    /** Panggil subprogram untuk membaca anggota himpunan 2 */
    inputAnggotaHimpunan(set2)

    output("Himpunan 1 = Himpunan 2?")
    /** Panggil subprogram untuk memeriksa kesamaan himpunan 1 dan
        himpunan 2, kemudian tampilkan hasilnya */
    if (himpunanSama(set1, set2)) then
        output("True")
    else
        output("False")
    endif
endprogram
```

4. Kemudian coba jalankan dengan **Build >> Build and Run** atau menekan **F9**. Beri masukan data seperti contoh yang diberikan pada soal.
5. Pahami setiap kode program yang diberikan.
6. Selesai.