



## Program Studi Teknik Elektro ITB

Nama Kuliah (Kode) : Praktikum Pemecahan Masalah dengan C (EL2208)  
Tahun / Semester : 2021-2022 / Genap  
Modul : 1 - Overview of C Language  
Hari, Tanggal Praktikum : Rabu, 16 Februari 2022

---

### Naskah Soal Praktikum

Pembuat Naskah: Harry Purba, Reynaldo Averill Adji Putra

#### Ketentuan:

1. Kerjakanlah satu dari dua soal berikut pada *template repository* yang anda peroleh ketika mengambil *assignment* di GitHub Classroom praktikum!
2. *Commit* yang dilakukan setelah sesi praktikum berakhir tidak akan dipertimbangkan dalam penilaian.
3. *Header* setiap *file* harus mengikuti format yang telah disediakan pada *file template repository*. *Header* yang tidak mengikuti format tersebut tidak akan dinilai.
4. Buku catatan laboratorium yang berisi alasan pemilihan soal, *flowchart*, dan *data flow diagram* dari solusi yang anda buat dikumpulkan ke tugas.stei.itb.ac.id paling lambat pukul 11.00 WIB satu hari kerja setelah sesi praktikum.
5. Solusi soal pertama harus dapat dikompilasi dengan perintah `make soal-01` dan menghasilkan *file executable* dengan nama `soal-01`. Demikian pula, soal kedua harus dapat dikompilasi dengan perintah `make soal-02` dan menghasilkan *file executable* dengan nama `soal-02`.
6. Bila diperlukan, sesuaikanlah isi *Makefile* yang tersedia pada *template repository* untuk memenuhi syarat kompilasi dan *file* keluaran di atas!

### Soal 1

Mbak Keqing adalah pelajar yang sedang mempelajari angka ganjil, genap dan prima. Oleh karena itu, Mbak Keqing ingin membuat program yang dapat menampilkan setiap bilangan ganjil, genap, atau prima (**pilihan ganjil, genap, atau prima adalah pilihan dari user**) secara **ascending (urutan naik)** dari bilangan bulat A hingga bilangan bulat B. (**A dan B adalah input dari User dengan syarat B harus lebih besar dari atau sama dengan A**). Setelah itu program menampilkan jumlah banyaknya bilangan ganjil, genap, atau prima yang ada.

#### Notes:

- Bilangan prima merupakan bilangan asli yang lebih besar dari 1 yang memiliki tepat dua faktor (bilangan asli) yaitu 1 dan dirinya sendiri.
- Angka 0 dianggap sebagai bilangan genap.
- Bilangan negatif bukan merupakan bilangan ganjil atau genap

#### Contoh Eksekusi Program (garis bawah menandakan input)

**#1**

PROGRAM PENCARIAN BILANGAN

Bilangan apa yang ingin dicari?

1 : Ganjil

2 : Genap

3 : Prima

Silahkan masukkan input : 0

Input salah. Program selesai!

**#2**

PROGRAM PENCARIAN BILANGAN

Bilangan apa yang ingin dicari?

1 : Ganjil

2 : Genap

3 : Prima

Silahkan masukkan input : 1

PROGRAM PENCARIAN BILANGAN GANJIL

Masukkan bilangan awal mulai : 3

```
Masukkan bilangan akhir : 2
Input salah. Masukkan input kembali!

Masukkan bilangan awal mulai : 2
Masukkan bilangan akhir : 3

Semua bilangan ganjil dari 2 sampai 3 adalah :
3

Banyaknya bilangan ganjil : 1
```

### #3

#### PROGRAM PENCARIAN BILANGAN

Bilangan apa yang ingin dicari?

- 1 : Ganjil
- 2 : Genap
- 3 : Prima

Silahkan masukkan input : 2

#### PROGRAM PENCARIAN BILANGAN GENAP

Masukkan bilangan awal mulai : -15

Masukkan bilangan akhir : 10

Semua bilangan genap dari -15 sampai 10 adalah :

```
0
2
4
6
8
10
```

Banyaknya bilangan genap : 6

### #4

#### PROGRAM PENCARIAN BILANGAN

Bilangan apa yang ingin dicari?

- 1 : Ganjil
- 2 : Genap
- 3 : Prima

Silahkan masukkan input : 3

#### PROGRAM PENCARIAN BILANGAN PRIMA

Masukkan bilangan awal mulai : 0

Masukkan bilangan akhir : 20

Semua bilangan prima dari 0 sampai 20 adalah :

2  
3  
5  
7  
11  
13  
17  
19

Banyaknya bilangan prima : 8

## #5

PROGRAM PENCARIAN BILANGAN

Bilangan apa yang ingin dicari?

1 : Ganjil  
2 : Genap  
3 : Prima

Silahkan masukkan input : 3

PROGRAM PENCARIAN BILANGAN PRIMA

Masukkan bilangan awal mulai : -25

Masukkan bilangan akhir : -10

Semua bilangan ganjil dari 0 sampai 20 adalah :

Banyaknya bilangan prima : 0

## Soal 2

Han baru saja mempelajari cabang baru dalam ilmu matematika, yaitu teori bilangan. Ia belajar bahwa setiap bilangan dapat memiliki representasi yang berbeda-beda. Di dalam kehidupan sehari-hari, basis bilangan yang digunakan adalah basis 10. Dalam dunia *engineering*, basis bilangan yang sering digunakan adalah basis 2 (sering disebut sebagai biner). Ia baru sadar bahwa setiap angka dapat menjadi bilangan yang berbeda, tergantung dari basisnya. Misalkan terdapat suatu angka  $(a_n a_{n-1} a_{n-2} \dots a_2 a_1 a_0)_m$  yang dituliskan dalam basis  $m$ , dengan seluruh angka  $a_n, a_{n-1}, a_{n-2}, \dots, a_1, a_0$  dan  $m$  adalah angka dalam basis 10 (angka 0 sampai 9). Misalkan  $x$  merupakan nilai bilangan tersebut dalam basis 10. Maka,  $x$  dinyatakan dengan persamaan:

$$x = m^n \times a_n + m^{n-1} \times a_{n-1} + \dots + m \times a_1 + m^0 \times a_0$$

Sebagai contoh, 17 dapat dituliskan juga menjadi  $21_8$  dalam basis 8 karena  $17 = 8 \times 2 + 8^0 \times 1$  atau  $122_3$  dalam basis 3 karena  $17 = 3^2 \times 1 + 3 \times 2 + 3^0 \times 2$ . Berdasarkan hal ini, Han mendapatkan ide untuk menciptakan sistem kode yang ia sebut sebagai *Han's Code*. Ia akan menggunakan ini untuk mengirimkan pesan angka kepada temannya. Aturan membaca *Han's Code* adalah sebagai berikut.

1.  $a_n a_{n-1} a_{n-2} \dots a_2 a_1 a_0$  merupakan angka yang akan diterjemahkan (disebut sebagai angka Han) dengan jumlah digit  $n + 1$  dan angka  $a_n, a_{n-1}, a_{n-2}, \dots, a_1, a_0$  adalah angka dalam basis 10 (angka 0 sampai 9), sedangkan  $m_k, m_{k-1}, \dots, m_1$ , dan  $m_0$  merupakan angka yang digunakan sebagai basis (disebut sebagai basis Han). Asumsikan bahwa  $k \leq n$ . Angka Han merupakan bilangan asli.
2. Basis yang mungkin digunakan pada kode ini adalah basis 2 hingga 9 (9 kemungkinan basis).
3. Pembacaan kode diawali dengan cara mencari representasi basis 10 dari  $a_n a_{n-1} a_{n-2} \dots a_2 a_1 a_0$  dengan digit  $a_n$  dibaca sebagai basis  $m_k$ ,  $a_{n-1}$  dibaca sebagai basis  $m_{k-1}$ , dan seterusnya. Apabila  $k < n$  sehingga  $m_0$  telah tercapai, maka pembacaan digit selanjutnya diulangi kembali dari basis  $m_k$  hingga seluruh digit telah tercapai. Anggap hasil pembacaan adalah  $y_z y_{z-1} \dots y_1 y_0$ . Sebagai contoh angka 12325 akan dibaca dengan basis 7, 2 dan 3 ( $m_2 = 7, m_1 = 2, m_0 = 3$ ). Maka, representasi basis 10 dari 12325 yang ditulis dalam basis 7, 2, dan 3 adalah  $7^4 \times 1 + 2^3 \times 2 + 3^2 \times 3 + 7^1 \times 2 + 2^0 \times 5 = 2401 + 16 + 27 + 14 + 5 = 2463$ . Maka,  $y_3 = 2, y_2 = 4, y_1 = 6$  dan  $y_0 = 3$ .

4. Setelahnya, angka yang diperoleh dari bagian 3 akan dibaca sebagai basis  $\{y_z, y_{z-1}, \dots, y_1, y_0, m_k, m_{k-1}, \dots, m_0\} = q$ . Pada kumpulan bilangan  $y_z, y_{z-1}, \dots, y_0$ , ganti seluruh elemen yang nilainya sama dengan  $q$  dengan  $q - 1$ . Kemudian, hasil akhir/*decode Han's Code* merupakan representasi basis 10 dari bilangan  $y_z y_{z-1} \dots y_1 y_0$  yang telah diganti sebelumnya dan dituliskan dalam basis  $q$ . Sebagai contoh, untuk basis hasil perhitungan 2463 dengan basis 7, 2, dan 3, nilai  $q = \{2, 4, 6, 3, 7, 2, 3\} = 7$ . Lalu, ganti seluruh digit  $q$  dengan  $q - 1$ , sehingga bilangannya menjadi 2, 4, 6, 3 (tidak ada 7 di antara  $y_3 y_2 y_1 y_0$ , sehingga bilangan tetap). Maka, hasil *decode/akhir* dari *Han's Code* merupakan representasi basis 10 dari  $2463_7$ , yaitu  $7^3 \times 2 + 7^2 \times 4 + 7 \times 6 + 3 = 927$ .

Bantulah Han untuk membuat sebuah program dalam bahasa C yang dapat digunakan untuk melakukan *decode* angka Han!. Program menerima *input* berupa angka Han, jumlah basis Han, serta masing-masing basis Han nya. Program memberikan *output* berupa hasil *decode* angka Han.

#### Contoh Eksekusi Program (garis bawah menandakan input)

##### #1

```
Masukkan angka Han: 12325
Masukkan jumlah basis Han: 3
Masukkan basis Han indeks ke 2: 7
Masukkan basis Han indeks ke 1: 2
Masukkan basis Han indeks ke 0: 3
Hasil decode angka Han adalah 927
```

##### #2

```
Masukkan angka Han: 1
Masukkan jumlah basis Han: 1
Masukkan basis Han indeks ke 0: 8
Hasil decode angka Han adalah 1
```

##### #3

```
Masukkan angka Han: 9000
Masukkan jumlah basis Han: 4
Masukkan basis Han indeks ke 3: 2
Masukkan basis Han indeks ke 2: 3
Masukkan basis Han indeks ke 1: 3
Masukkan basis Han indeks ke 0: 2
Hasil decode angka Han adalah 44
```

**#4**

Masukkan angka Han: 11111111

Masukkan jumlah basis Han: 2

Masukkan basis Han indeks ke 1: 3

Masukkan basis Han indeks ke 0: 8

Hasil decode angka Han adalah 94197