

Latihan Problem Solving dengan Membuat Program-program Sederhana

Sebelum Anda menuliskan sebuah program, Anda perlu membuat Spesifikasi.

Spesifikasi program adalah kalimat ringkas yang menyatakan:

1. Input yang diberikan pengguna program.
2. Komputasi yang dilakukan (sekuensial, kondisional, pengulangan). jika terdiri dari beberapa tahap, maka dijelaskan dalam urutan instruksi berdasarkan *construct* program yang sudah dijelaskan.
3. Output yang dihasilkan program.

Anda dapat (dan sangat dianjurkan) untuk mengambil salah satu program yang pernah dibahas sebagai pola solusi, dan menjelaskan perbedaannya dengan program yang Anda hasilkan. Inilah Computational Thinking!

Pada bagian ini diberikan sebuah contoh-contoh program sederhana, yang dapat Anda pakai sebagai bahan belajar untuk membangun solusi mulai dari sederhana sampai dengan yang lebih kompleks. Jika ada menghadapi permintaan sebuah program kompleks, Anda perlu untuk memahami abstraksinya, dan mendekomposisi menjadi persoalan-persoalan yang lebih "melakukan satu hal", dan mengimplementasi berdasarkan pola solusi yang Anda punyai.

I. Komputasi berdasarkan Rumus

Contoh berikut adalah contoh program perhitungan sederhana:

```
5. // contoh program menghitung luas lingkaran
6. #include<stdio.h>
7. int main ()
8. { /* Kamus */
9.     const float pi = 3.1415;
10.    float r;
11.    /* program */
12.    /* baca data */
13.    printf ("Jari-jari lingkaran =");
14.    scanf ("%f", &r);
15.    /* Hitung dan tulis hasil */
16.    printf ("Luas lingkaran = %f\n", pi * r * r);
17.    printf ("Akhir program \n");
18.    return 0;
19. }
```

Latihan soal kreatif

Berdasarkan contoh program komputasi sederhana untuk menghitung luas lingkaran, Anda dapat menuliskan spesifikasi lebih jelas untuk menyelesaikan persoalan berikut, mulai dari yang paling sederhana dan mirip, kemudian mengembangkan menjadi program yang lebih kompleks karena tidak hanya menyelesaikan satu persoalan saja, tetapi menyelesaikan beberapa persoalan sekaligus.

1. JARAK. Tuliskanlah program yang:

1. Membaca nilai v (kecepatan), dan t (waktu tempuh) sebuah mobil yang bergerak lurus beraturan dari kota A ke kota B.
2. Menghitung dan menuliskan jarak kota A ke kota B, yang ditempuh mobil tersebut.

2. LUASPERSEGI. Tuliskanlah program yang membaca panjang dan lebar sebuah lantai ruangan yang berbentuk persegi (dalam m), kemudian menghitung luas lantai tersebut dalam m^2 .

3. KUBUS. Tuliskanlah sebuah program dalam bahasa C yang membaca sisi kubus (dalam cm), dan menghitung serta menuliskan volumenya.

4. Anda dapat menuliskan program kalkulasi apa pun berdasarkan rumus yang sudah ditentukan [buatlah latihan mandiri, minimal dari 50 rumus geometri, fisika, kimia, pertumbuhan makhluk secara eksponensial dalam biologi, rumus bunga berbunga dalam bidang ekonomi].

5. Untuk perhitungan berdasarkan rumus, Anda dapat melakukan generalisasi, dan menuliskan sebuah program yang akan memberikan pilihan kepada pengguna, untuk:

1. Memilih bentuk geometris segitiga, persegi, bujur sangkar, belah ketupat, jajaran genjang, segi N-beraturan, elipse, lingkaran.
2. Menentukan ciri dari bentuk geometris yang sudah dipilih, dan meminta pengguna memasukkan data besaran untuk menghitung luas dan kelilingnya (misalnya untuk segitiga, alas dan tingginya, ~~sebab luas~~ ditentukan oleh tinggi segitiga, sedangkan keliling ditentukan oleh panjang ketiga sisinya).
3. Melakukan perhitungan sesuai rumus.

6. Untuk sebuah bentuk geometris tiga dimensi, Anda dapat menulis sebuah program untuk membaca cirinya dan menghitung luas permukaan dan, volumenya.
7. Setelah program pada (5) dan (6) berjalan dengan baik, Anda dapat menambahkan pengulangan sehingga program dapat dipakai berulang (tidak hanya satu kali komputasi langsung berhenti).

1. Program berhenti setelan N kali dipakai pengguna.

2. Program berhenti setelah pengguna mengetikkan pilihan tertentu, yang bukan merupakan pilihan bentuk geometris yang diberikan.

II. Latihan Kondisional

- Berdasarkan contoh program kecil untuk menentukan bilangan negatif, nol dan positif, tuliskan spesifikasi program sebagai berikut, dan kodelah programnya dalam bahasa C:
1. AIR - Sebuah program akan membaca T, yaitu temperatur air pada tekanan atmosfir, dan menentukan apakah air tersebut dalam keadaan beku, padat atau uap. Tentukan angka-angka yang diperlukan untuk mengetes program ini.
2. BERATBADAN - Program yang akan membaca berat badan seseorang, dan menentukan apakah termasuk Kurus, Ideal, atau Gemuk, Kegemukan.


1. Buatlah aturan yang masuk akal, untuk menentukan berat badan yang dikategorikan Kurus, Ideal, atau Gemuk, Kegemukan.

2. Kriteria kategori tersebut ternyata berbeda untuk anak-anak/dewasa; pria/wanita. Buatlah untuk masing-masing, aturannya dan modifikasi programnya dengan membaca umur orang, jenis kelaminnya, baru menentukan berdasarkan aturan yang Anda buat.
3. BAJU - Sebuah program yang akan membaca ukuran panjang baju, lebar bahu, dan akan menentukan apakah pemilik ukuran tersebut harus diberi baju S (small), M (medium), atau L (large). Buatlah aturannya yang masuk akal, dan Anda harus menjamin bahwa setiap data akan diasosiasikan ke salah satu ukuran baju walaupun tidak sama persis.

III. Latihan Loop - Pengulangan

1. Tuliskanlah sebuah program yang akan mencetak berbagai output sebagai berikut, tergantung nilai N yang dibaca. N dapat berharga antara 1 s.d. 1000.

N=1	N=2	N=3	N=4
*	*	*	*
	**	**	**
		***	***

 DIBANTU

N=1	N=2	N=3	N=4
*	**	***	****
	*	**	***
		*	**
			*

3. Tuliskanlah sebuah program yang akan mencetak berbagai output sebagai berikut, tergantung nilai N yang dibaca. N dapat berharga antara 1 s.d. 1000.

N=1	N=2	N=3	N=4
o	oo	ooo	oooo
	oo	o*o	o**o
		ooo	o**o
			oooo

4. Dengan mengambil pola program dari contoh-contoh loop, tuliskan program yang akan menuliskan lagu anak ayam, dimulai dari:

Anak ayam turun 10; Mati 1 tinggal 9.

Anak ayam turun 9; Mati 1 tinggal 8.

...

Anak ayam turun 1; Mati 1 tinggal induknya.

5. Tuliskanlah sebuah program yang membaca sebuah nilai N, kemudian menuliskan hasil penjumlahan deret tambah 1+2+3+4+5+6+ . . . +N= XXXX.

Contoh untuk berbagai nilai N:

N=1	N=2	N=3	N=4
1=1	1+2=3	1+2+3=6	1+2+3+4=10

6. Tuliskanlah sebuah program yang membaca sebuah nilai N, kemudian menuliskan hasil penjumlahan deret tambah dengan suku bilangan ganjil saja 1+3+5+7+ . . . +N= XXXX.

Program harus dapat diulang untuk berbagai nilai N, sampai pengguna mengetikkan N=0 tanda berhenti.

N=1	N=2	N=3	N=4
1=1	1+3=4	1+3+5=8	1+3+5+7=16

7. Tuliskan sebuah program yang membaca sebuah nilai N, kemudian menuliskan hasil penjumlahan:

1. Deret : 1 +1/2 + 1/3 + ... +1/N
2. Deret : 1 - 1/20 + 1/30 - 1/40 ... 1/(N*10)

8. Tuliskan sebuah program yang melakukan pengulangan untuk menghitung faktorial (N) : 1x2x3x... N dan cetaklah N terbesar yang hasilnya masih lebih kecil dari 3267 (nilai maksimal yang dapat disisr sebuah variabel integer).



10. Tuliskan sebuah program untuk mencetak berbagai deret yang ada di

<https://en.wikipedia.org/wiki/Sequence> atau pada <https://www.mathsisfun.com/algebra/sequences-series.html> atau sumber lainnya.

Untuk membuat latihan lebih seru, setiap program yang mengandung pengulangan di atas dapat Anda ubah sehingga dapat diulang sesuai kriteria yang Anda tentukan. Ini Computational Thinking!

IV. Latihan Tabel

Untuk sekumpulan nilai numerik yang sudah dibaca dan disimpan dalam tabel:

1. Tuliskan minimal 8 macam algoritma sorting dan 8 algoritma searching, dengan melakukan eksplorasi contohnya di Internet. Bandingkanlah algoritmanya, dan jelaskan mana yang lebih efisien.
2. Anggaplah sebuah tabel menyimpan satu kolom data pada sebuah spreadsheet, misalnya excel. pelajari beberapa fungsi excel, dan tuliskan fungsi/prosedur untuk mengimplementasi beberapa fungsi yang ada di Excel (eksplorasi), seperti:
 1. Untuk mengolah data bertipe integer count, countif, sum, sumif, max, min, average, standar deviasi, dll.
 2. Untuk menampilkan data dalam berbagai format, misalnya data angka menjadi berbagai format akunting, currency, tanggal.

V. Latihan Fungsi dan Prosedur

1. Tuliskan ulang program-program yang telah Anda implementasikan, dengan mengelompokkan instruksi menjadi fungsi dan/atau prosedur yang sesuai dengan konteksnya.
2. Untuk beberapa fungsi dan prosedur yang "mirip", tuliskan ulang menjadi fungsi/prosedur yang diparameterisasi sesuai dengan konteks dan penggunaannya.

[< Sebelumnya](#)[Selanjutnya >](#)

click bila
belum muncul

click bila
belum muncul