

Roadmap Belajar Pemrograman dari Kabupaten ke Nasional

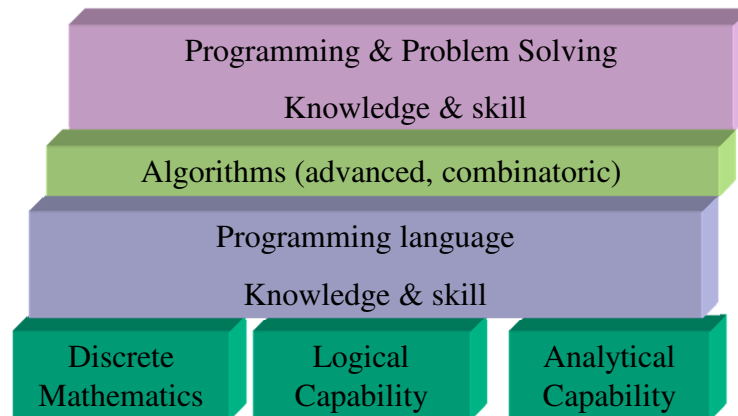


Inggriani Liem
inge@informatika.org
TOKI Biro ITB

Tujuan Presentasi

- Memberikan gambaran tentang apa yang harus dikuasai, yang harus dilakukan para murid agar mampu “memprogram” dalam bahasa Pascal yang disyaratkan TOKI
- Persamaan dan perbedaan antara mengajar programming di perguruan tinggi dengan mengajar programming untuk TOKI
- Sharing mengenai Tips mengajar, membina siswa dengan mengenali karakteristik kandidat TOKI

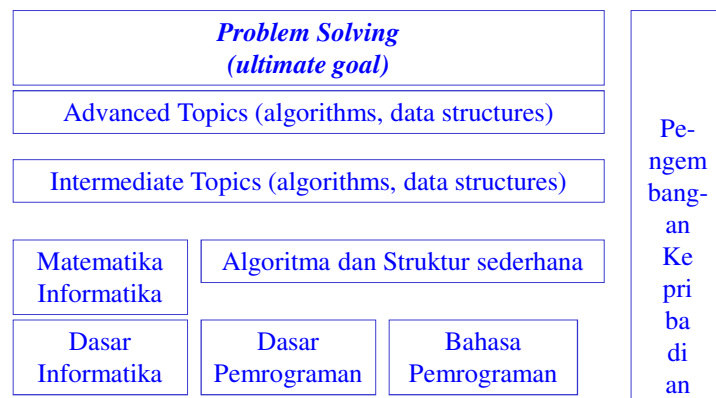
TOKI Requirement



Inggriani Liem, TOKI Biro ITB-2009

3

Susunan materi Mulai 30 besar



Inggriani Liem, TOKI Biro ITB-2009

4

Harapan Tingkatan Pencapaian

- **Kabupaten dan Propinsi:**
 - Kemampuan analitika, logika, aritmetika
 - Kemampuan algoritmika (programming)
- **Nasional:**
 - Memahami soal
 - Memahami input/output
 - Memprogram, untuk menyelesaikan persoalan “sederhana” [dalam bahasa Pascal]
- **Internasional (TOKI) :** menyelesaikan persoalan-persoalan (*problem solving*) dengan memilih solusi yang optimal, akan dicapai dalam 3 tahap yang menaik: 30
~~besar, 15 besar, 8 besar.~~

Inggrani Liem, TOKI Biro ITB-2009

5

Catatan

- Daya logika, analisis dan dasar matematika yang kuat diperlukan karena merupakan “potensi” dan dasar untuk mampu memprogram
- Oleh karena itu, bagi yang belum banyak mengenal komputer/memprogram, test mengenai pengetahuan dasar tersebut dapat memunculkan “bakat tersembunyi”

Inggrani Liem, TOKI Biro ITB-2009

6

Aktivitas Siswa dalam Programming

- Main-main, mencoba-coba, untuk memahami persoalan, dan menghasilkan solusi.
- Program Design: analisis, desain program *[bisa tanpa komputer]*
- Program Writing: coding, debugging & testing *[harus dengan komputer]*
- Program Reading : mempelajari teks program *[dengan/tanpa komputer]* atau mengamati eksekusi program *[harus dengan komputer]*
- Menghitung kompleksitas dan membuktikan kebenaran program *[secara matematis, tanpa komputer]*

Inggriani Liem, TOKI Biro ITB-2009

7

Bahasa Pemrograman versus Pemrograman

- Belajar Bahasa pemrograman : sintaks (tatabahasa), elemen bahasa.
- Belajar pemrograman : memakai salah satu bahasa pemrograman untuk menyelesaikan suatu persoalan *[dengan komputer]*
- Belajar bahasa pemrograman lebih mudah daripada belajar pemrograman (*problem solving*)

Inggriani Liem, TOKI Biro ITB-2009

8

Elemen Program [Prosedural]

- Program = Struktur Data + Algoritma (instruksi)
- Struktur data : dasar (int, real, boolean), bentukan (record, array, set)
- Instruksi : assignment, read/write, if/case, loop (for, while, repeat)
- Pengelompokan instruksi menjadi fungsi/prosedur
- Operasi file eksternal.

“Elemen” program vs Program “Utuh”

- Harus ada program Utama dan dapat dieksekusi.
- Semua deklarasi dan kalimat program mempunyai “konteks” (program utama, sub program)
- Oleh karena itu, setiap “potongan” program harus jelas konteksnya
- Program disusun setelah dipikirkan/dianalisis solusi optimalnya.

Bahasa Pascal

- Diciptakan oleh Niklaus Wirth, salah seorang tokoh di bidang pemrograman

Para Guru disarankan membaca buku Niklaus Wirth (spesifikasi bahasa Pascal, walaupun mungkin sulit dipahami dibandingkan dengan buku-buku yang tersedia di pasaran)

- Bahasa pemrograman yang relatif “sederhana”, mudah dikuasai
- Oleh karena itu Pascal dipilih sebagai bahasa tim Indonesia

Important Pascal Keywords & Operator

program

TYPE
CONST
VAR

procedure

function

begin
end

integer

real

boolean

record

set of

array

file of ...

* / + - div mod

= < > <= >= <>

and or not xor

:=

^ @ . with

read, readln

write, writeln

if..

If.. then...

if ..then..else

case ... of..

for ... to.. do..

for ... downto.. do..

while .. do ..

repeat .. until...

Perbedaan ‘pengajaran’ programming TOKI vs Kuliah di PT

- Pada Kuliah, mata kuliah programming seringkali merupakan mata kuliah wajib dan biasanya [kecuali di beberapa PT], di PT jam terbang kurang (terutama untuk S1). Pada pelatihan TOKI, jam terbang harus tinggi.
- Pada kuliah, design dan spesifikasi harus diformalkan untuk menjadi bagian dari deliverables. Untuk lomba TOKI, source code satu-satunya deliverables yang disetor

Perbedaan ‘pengajaran’ programming TOKI vs Kuliah di PT

- Struktur program, pemilihan instruksi dari bahasa pemrograman, struktur data “abstrak” di perkuliahan merupakan parameter penilaian. Di lomba, hal ini tidak diperhatikan walau mungkin dapat memberikan benefit kepada ybs.
- Di perkuliahan, dosen kurang peduli bagaimana mahasiswa mengetik, menghasilkan program. Untuk lomba, faktor pengerjaan dan strategi menyelesaikan soal perlu diperhatikan.

Perbedaan ‘pengajaran’ programming TOKI vs Kuliah di PT

- Program quality (readability, structure, robustness, **performance**), hanya sebagian yang penting di lomba TOKI, karena programnya bukan untuk dipakai, dipelihara ☺ ☺ ☺. Tapi, beberapa hal perlu ditekankan untuk menang!
- Performance (waktu, memori) yang di kuliah kurang diperhatikan, justru di TOKI menjadi sangat penting

Perbedaan ‘pengajaran’ programming TOKI vs Kuliah di PT

- Testing dan test case : di kuliah programming, kurang dipentingkan dan biasanya program dinilai manusia. Di lomba TOKI, karena auto grader bekerja berdasarkan test case, maka pemahaman mengenai test case menjadi penting.

Perbedaan ‘pengajaran’ programming TOKI vs Kuliah di PT

- Soal-soal di kuliah programming biasanya merupakan soal standard untuk solusi persoalan nyata. Di TOKI, ada aspek “fun”, senang, bermain, soal-soal dikemas dalam konteks tertentu sehingga lucu.
- Persoalan programming di kuliah PT Indonesia, khususnya yang sistem informasi, dan pengolahan data, yang justru merupakan persoalan yang sering menjadi topik di PT, justru bukan type soal-soal IOI. Soal-soal IOI lebih ke arah “algoritma lanjut” di bidang Computer Science.

Karakteristik Siswa

- Pada IOI, dilombakan “problem solving” dengan membuat program.
- Komputer (programming) bukan mata pelajaran wajib seperti bidang olimpiade yang lain
- Biasanya, pelajaran komputer diasosiasikan ke pelatihan memakai komputer (perangkat lunaknya), khususnya aplikasi office. Mungkin sekarang ini sudah menjadi wajib
- Biasanya, pelajaran programming bukan pelajaran wajib, dan sejujurnya, kemampuan programming untuk problem solving belum terlalu dibutuhkan untuk rata-rata lulusan SMA

Karakteristik Siswa Kandidat TOKI

- Senang “bermain” dengan komputer
- Tertarik untuk memrogram
- Kemampuan bahasa Inggrisnya memadai
- Mampu belajar sendiri [agak “menyendiri”, nyentrik]
- Guru dan kepala sekolah yang mengenal mereka dalam sehari-hari, yang dapat mengenali kandidat
- Ada satu dua kemampuan sangat menonjol dibandingkan yang lain, perlu dibina secara khusus

Saran Untuk Siswa

- Belajar & berlatih dengan disiplin, rutin, tekun.
- Kemampuan programming hanya dapat dikuasai oleh diri sendiri, bukan oleh orang lain.
- Rajin membaca buku-buku programming dan mencoba mengimplementasi semua algoritma yang dibahas ke dalam bahasa [Pascal] sampai dipahami benar kinerjanya.
- Latihan soal secara rutin dari sumber-sumber berkaitan lomba programming.

Saran Untuk Siswa [2]

- Aktif di milis toki:
- Mengetahui jadwal, berita terakhir dan kegiatan-kegiatan relevan dengan toki
- Membahas solusi yang pernah dibuatnya (untuk perbaikan) dan bertanya jika tidak mengerti
- Mengikuti situs-situs lomba sejenis.
Untuk persiapan OSN: usaco contest level-1:
 - <http://ace.delos.com/ioigate>

Saran Untuk Siswa [3]

- Siswa sangat disarankan untuk bergabung dengan milis toki : tokinet@yahoo.com
- Anggota milis tsb : pembina toki, alumni toki, toki-ers, siswa SMA, guru... siapapun.
- Diskusi di milis tersebut dapat membantu siswa untuk menambah pengetahuannya.
- Cukup banyak alumni TOKI yang aktif mengarahkan adik-adiknya.
- Ada pula diskusi mengenai persoalan-persoalan umum dan 'kepenasaran' siswa tentang berbagai hal

Persiapan Siswa Untuk Mengikuti OSP

- Pada saat mengikuti OSP, siswa seharusnya sudah mampu
 - Kemampuan analitik: menjawab soal-soal analitik tingkat cukup tinggi
 - Kemampuan algoritmik
 - Membaca pseudocode (program dalam pseudo pascal)
 - Memahami aspek eksekusi
- Walaupun test dilaksanakan **tanpa komputer**, disarankan bahwa siswa sudah mulai memprogram karena aspek eksekusi lebih mudah dipahami dengan menjalankan program dan “bermain” dengan kompiler dan komputer.
- Memahami dan menguasai bahan-bahan yang ada di situs <http://www.toki.or.id>

Setelah Lulus OSP

- Setelah lulus OSP, perlu persiapan untuk mengikuti OSN,
- Segera setelah OSP, siswa harus rajin dalam mengikuti PJJ (Pelatihan Jarak Jauh) dari <http://www.toki.or.id> :
 - Registrasi dengan referensi sekolah/guru, supaya guru ikut memonitor
 - Aktif berlatih
- Dengan mengikuti PJJ, siswa mempunyai kesempatan berlatih menggunakan aplikasi TOKI Online Judge yang dipakai saat OSN (lihat slides manual ringkas yang ditulis terpisah).

Persiapan OSN

- Antara OSP dan OSN, ada perbedaan:
 - Cara ujian : OSP tanpa komputer, OSN sepenuhnya menggunakan komputer
 - Tingkat kesulitan
- Perhatikan kemampuan yang diharap sudah dikuasai murid sebelum maju ke OSN.
- Hubungi Toki Pusat/Toki Biro/Perguruan Tinggi terdekat jika memerlukan bantuan.

Persiapan Menuju OSN [1]

- Semua lomba pada saat OSN dilaksanakan dengan menggunakan komputer.
- Oleh karena itu Siswa harus mempunyai kemampuan minimum menggunakan komputer, yaitu sudah tidak canggung untuk:
 - Mengoperasikan komputer dengan sistem operasi standard.
 - Berinteraksi mengisi/menjawab via aplikasi berbasis Web (lihat screen shot aplikasi ...).
 - Membuka/mengisi/berpindah folder.
 - Menyimpan/mendownload file.

[*Detail*](#)

Alamat Website

- TOKI Pusat
 - <http://www.toki.or.id>
- TOKI Biro
 - ITB: <http://toki.if.itb.ac.id>
 - IPB: <http://toki.ipb.ac.id>
 - UGM: <http://cs.ugm.ac.id>
 - ITS : <http://if.its.ac.id>

Tips [1]

- Awali dengan membangun perilaku “senang” memprogram ke anak-anak
- Sedapat mungkin, guru pernah memprogram
- Ajarkan type data/instruksi sesuai kebutuhan
- Bedakan belajar “mengetik” prgram dengan belajar memprogram.
- Program reading, sama pentingnya dengan program writing. [membaca cepat]
- Ajarkan pemrograman sebagai bagian kegiatan ko-kurikuler (pisahkan antara programming dengan pemakaian aplikasi office)

Tips ... [2]

- Untuk tingkat kabupaten, Disediakan sebuah dokumen tipis untuk Guru dan Murid (Program kecil dalam bahasa Pascal yang sangat mirip dengan algoritmik supaya siswa bisa latihan sendiri dengan komputer) yang dapat dipakai oleh pemula untuk *memulai* membuat program. Selanjutnya, dapat mem-variasi, mengkomposisi, mengkombinasi instruksi/struktur data untuk membuat program besar.
- Tips mengajar Pascal sebagai bahasa pertama (bagi Guru, dosen).

Catatan Akhir

- Bahasa pemrograman, dan bisa memrogram hanya merupakan salah satu syarat untuk berhasil dalam IOI. Fokusnya problem solving.
- Lebih ke arah Computer Science daripada Information technology secara umum, atau pengolahan data dan sistem informasi
- Tantangannya bagaimana mengenali siswa yang berminat, dan berbakat, untuk dididik secara khusus.

Definisi Kompetensi

Persyaratan Kemampuan Bekerja dg Komputer saat OSN

- Melakukan “log in” untuk mulai bekerja dengan komputer
- Mengetik dengan keyboard
- Memakai mouse
- Membuka/mengisi/berpindah folder
- Menyimpan/menimpa file

Persyaratan Kemampuan Interaksi Dengan Aplikasi Web

- Mengakses sebuah URL.
- Membaca teks yang dituliskan.
- Mengenali tombol navigasi dan interaksi, serta area-area penting dalam sebuah halaman web.
- Memilih pilihan jawaban (radio button)
- Mengisi dengan teks.
- Melakukan submission (upload file).

Persyaratan Kemampuan Pascal Programming [OSN]

- Mengenal sintaks Pascal
- Mengenal, mampu memakai dan memahami aspek eksekusi :
 - Type primitif dan variannya, VAR, CONST
 - Assignment (:=)
 - Standard input/output (membaca/tulis) dari/ke file teks
 - Instruksi kondisional (if..then..; if..then..else; case...)
 - Loop (for ...; while...; repeat)
 - Prosedur dan Fungsi, parameter passing
 - Struktur data dasar (array, matriks)

Persyaratan Kemampuan Problem Solving

- Siswa harus sudah mampu menyelesaikan persoalan (problem solving) dengan membuat program Pascal
- Lihat contoh-contoh soal yang ada di situs <http://www.toki.or.id> untuk mengetahui tingkat kesulitannya.

Bekerja Dengan Kompilator Pascal

- Mengedit, menyimpan source code
- Mengkompilasi sebuah source code
- Mengamati hasil eksekusi di jendela “output”
- Memahami format output, batasan memori dan waktu dari deskripsi soal