

BUKU PEMBAHASAN

Bebras Indonesia

**COMPUTATIONAL THINKING
CHALLENGE 2019**

PENEGAK
(Tingkat SMA/MA
SMK/MAK)



<https://bebras.or.id/>

Diterbitkan oleh : NBO Bebras Indonesia

Pengantar

Tantangan Bebras Indonesia adalah kompetisi yang dilaksanakan secara online dan serentak dengan memberikan soal-soal yang telah dipersiapkan dalam Workshop Bebras Internasional, pada periode Bebras Week di minggu pertama/dua bulan November.

Tantangan Bebras Indonesia dibagi dalam 3 kategori, yaitu:

1. Siaga, untuk siswa SD/MI dan yang sederajat
2. Penggalang, untuk siswa SMP/MTs dan yang sederajat
3. Penegak, untuk siswa SMA/MA/SMK/MAK dan yang sederajat.

Untuk kategori Siaga (SD/MI) diberikan 10 soal yang harus diselesaikan dalam waktu 40 menit. Untuk kategori Penggalang (SMP/MTs) dan Penegak (SMA/MA/SMK/MAK) masing-masing diberikan 12 soal yang harus diselesaikan dalam waktu 45 menit.

Tantangan Bebras Indonesia 2019 dapat berjalan lancar berkat dukungan penuh dari IPB yang menyediakan dan mengelola <https://tantanganbebras.ipb.ac.id> dan GDP Labs yang menyediakan dan mengelola <https://olympia.id> sebagai sistem aplikasi untuk lomba online. Selain dari itu juga LAPI Divusi yang membantu mengelola situs <https://bebras.or.id> serta para Koordinator Bebras Biro dan tim yang tersebar di 60 perguruan tinggi di seluruh Indonesia yang langsung berhubungan dengan para siswa dalam menyelenggarakan Tantangan Bebras Indonesia 2019.

Penyiapan soal-soal dan pengelolaan Tantangan Bebras Indonesia 2019 dilaksanakan oleh Tim Olimpiade Komputer Indonesia (TOKI), yaitu: Inggriani (ITB), Adi Mulyanto (ITB), Suryana Setiawan (UI), Julio Adisantoso (IPB), Rully Soelaiman (ITS). Yudhi Purwananto (ITS), Yugo K. Isal (UI), dan Fauzan Joko Sularto (UPJ). Penyiapan soal juga dibantu oleh Mewati Ayub (UKM), Husnul, Natalia, dan Vania Natali (Unpar), serta penyiapan buku ini dibantu oleh Devi Indriasari (UAJY), Irya Wisnubhadra (UAJY), dan Risna Saragih

Bahan belajar Computational Thinking Tantangan Bebras Indonesia 2019 ini dibagi dalam tiga buku sesuai kategori, yaitu buku untuk Tingkat SD/MI (Siaga), Tingkat SMP/MTs (Penggalang), dan Tingkat SMA/MA/SMK/MAK (Penegak).

Karya ini dilisensikan di bawah lisensi Creative Commons Attribution-Non Commercial-No Derivatives 4.0 International (CC BY-NC-SA 4.0)

Hal ini berarti Anda bebas untuk menggunakan dan mendistribusikan buku ini, dengan ketentuan:

- Attribution: Apabila Anda menggunakan materi-materi pada buku ini, Anda harus memberikan kredit dengan mencantumkan sumber dari materi yang Anda gunakan.
- Non-Commercial: Anda tidak boleh menggunakan materi ini untuk keperluan komersial, seperti menjual ulang buku ini.
- ShareAlike: Apabila Anda mengubah atau membuat turunan dari materi-materi pada buku ini, Anda harus menyebarkan kontribusi Anda di bawah lisensi yang sama dengan materi asli.

Computational Thinking

Kemampuan berpikir kreatif, kritis dan komunikasi serta kolaborasi adalah kemampuan yang paling penting dalam (*21st century learning*) pembelajaran di abad kedua-puluh-satu, di antara kemampuan-kemampuan lainnya seperti membaca, matematik, sains. Siswa zaman sekarang perlu untuk mengembangkan keterampilan berpikir, menguasai pengetahuan tentang konten dari persoalan yang dihadapi (*content knowledge*), dan mempunyai kompetensi sosial dan emosional untuk mengarungi kehidupan dan lingkungan kerja yang semakin kompleks.

Di bidang “*Computing*” (diterjemahkan ke dalam bahasa Indonesia menjadi “Informatika”), kemampuan berpikir yang perlu dikuasai sejak pendidikan dasar adalah “*Computational Thinking*” (CT). CT adalah proses berpikir untuk memformulasikan persoalan dan solusinya, sehingga solusi tersebut secara efektif dilaksanakan oleh sebuah agen pemroses informasi (“komputer”, robot, atau manusia). CT adalah sebuah metoda dan proses berpikir untuk penyelesaian persoalan dengan menerapkan:

- **Dekomposisi** dan formulasi persoalan sedemikian rupa sehingga dapat diselesaikan dengan cepat dan efisien serta optimal dengan menggunakan komputer sebagai alat bantu.
- **Abstraksi**, yaitu menyarikan bagian penting dari suatu permasalahan dan mengabaikan yang tidak penting, sehingga memudahkan fokus kepada solusi.
- **Algoritma**, yaitu menuliskan otomatisasi solusi melalui berpikir algoritmik (langkah-langkah yang terurut).
- **Pengenalan pola persoalan**, generalisasi serta mentransfer proses penyelesaian persoalan ke sekumpulan persoalan sejenis.

CT adalah sebuah cara berpikir untuk memecahkan persoalan, merancang sistem, memahami perilaku manusia. CT melandasi konsep informatika. Di dunia saat ini dimana komputer ada di mana-mana untuk membantu berbagai segi kehidupan, CT harus menjadi dasar bagaimana seseorang berpikir dan memahami dunia dengan persoalan-persoalannya yang semakin kompleks. CT berarti berpikir untuk menciptakan dan menggunakan beberapa tingkatan abstraksi, mulai memahami persoalan sehingga mengusulkan pemecahan solusi yang efektif, efisien, “fair” dan aman. CT berarti memahami konsekuensi dari skala persoalan dan kompleksitasnya, tak hanya demi efisiensi, tetapi juga untuk alasan ekonomis dan sosial.

Bagaimana Belajar Computational Thinking?

Berpikir itu dapat dipelajari dan diasah dengan berlatih, serta mengkonstruksi pola pikir berdasarkan pengalaman. *Computational Thinking* juga dapat dipelajari dengan cara berlatih menyelesaikan persoalan-persoalan yang terkait komputasi, melalui persoalan sehari-hari. Lewat latihan-latihan yang menarik, siswa menerapkan teknik yang cocok (dekomposisi, abstraksi, pengenalan pola, representasi data, algoritmik) untuk mendapatkan solusi. Setelah latihan, siswa diharapkan melakukan refleksi serta mengkonstruksi pengetahuan berpikir, kemudian membentuk pola berpikir komputasional, yang semakin lama semakin tajam, cepat, efisien, dan optimal.

Apa perbedaan ICT/TIK (Teknologi Informasi dan Komunikasi) dengan Informatika?

Sejalan dengan itu, ICT (*Information and Communication Technology*, dalam bahasa Indonesia disebut Teknologi Informasi dan Komunikasi/TIK) mulai dibedakan dengan Informatika. TIK mengarah ke penggunaan teknologi dan perangkat/gadget, sedangkan Informatika mengarah ke keilmuan dan desain produk-produk informatika baik yang nyata (piranti pintar), maupun yang abstrak seperti program aplikasi, dan algoritma.

Kemampuan TIK lebih mengarah ke penggunaan teknologi dan perangkat/*gadget*, sedangkan Informatika mengarah ke keilmuan komputasinya. Penggunaan TIK yang dimaksud bukan hanya keterampilan menggunakan gadget dan aplikasinya, tetapi juga kemampuan untuk menggunakan dan

memanfaatkan konten dengan bijak, terutama di saat kecerdasan buatan sudah banyak digunakan untuk menciptakan konten digital.

Agar bangsa Indonesia mampu bersaing dengan negara lain, anak Indonesia tidak cukup menjadi pengguna teknologi saja, melainkan harus lebih kreatif dan inovatif untuk menciptakan produk-produk TIK. Untuk ini, siswa perlu mempelajari informatika.

Kaitan Computational Thinking dengan Informatika

Di negara maju, “*Computer Science*” (yang di Indonesia diterjemahkan sebagai “Informatika”) sudah mulai diajarkan sejak usia dini di tingkat pendidikan dasar, dengan materi dan kegiatan yang dirancang dengan mengacu ke kerangka kurikulum yang disusun oleh persatuan guru-guru, asosiasi profesi informatika, perusahaan terkemuka di bidang informatika dan TIK, serta organisasi-organisasi nirlaba yang peduli terhadap perlunya edukasi tentang informatika sejak usia dini [<https://k12cs.org>].

Kerangka kurikulum Informatika tersebut mendefinisikan lima bidang pengetahuan yaitu:

- Sistem Komputer (SK),
- Jaringan Komputer (JK),
- Analisis Data (AD),
- Algoritma dan Pemrograman (AP), dan
- Dampak Sosial Informatika (DSI).

Selain pengetahuan, juga didefinisikan praktik-praktik komputasi untuk mengemas pengetahuan dan mempraktikkannya, yaitu:

- pembinaan menumbuhkan budaya komputasi,
- menciptakan artefak berkolaborasi untuk mewujudkan suatu produk TIK,
- menguji dan memperbaiki/menyempurnakan artefak TIK,
- mengenali dan mendefinisikan problem-problem komputasi,
- berkomunikasi tentang komputasi, dan
- mengembangkan serta menggunakan abstraksi.

Sejalan dengan perbaikan yang berkelanjutan, dilakukan revisi terhadap organisasi elemen pembelajaran, yang dicerminkan dari perubahan lambang kerangka kurikulum Informatika Kurikulum Merdeka sejak Tahun 2024. Perubahan yang utama adalah dalam pengemasan elemen dan praktik yang lebih menekankan kepada pembelajaran informatika tanpa komputer (“*unplugged*”, *csunplugged*, <https://www.csunplugged.org/en/>), mengingat di Indonesia masih terjadi *digital gap* dan penekanan kepada pentingnya *Computational Thinking* sebagai landasan Informatika, dan sekaligus menjadi salah satu elemen penting pembelajaran abad ke-21 yang menjadi sasaran Profil Pelajar Pancasila. *Computational Thinking* seharusnya diperkenalkan secara intuitif dan menyenangkan sejak siswa masih pada jenjang SD, sedangkan Informatika menjadi mata pelajaran wajib dimulai dari SMP. Untuk ini Bebras berusaha menjembatani dengan membuat tantangan Bebras untuk siswa SD sebagai solusi, selain penguatan *Computational Thinking* yang khusus dibutuhkan dalam disiplin ilmu Informatika.

Makna lambang Elemen Informatika yang telah direvisi:

Lambang elemen Informatika ditunjukkan pada Gambar 1 Lambang Kurikulum Informatika yang terdiri dari Lambang Elemen Informatika sebelum direvisi dan Lambang Elemen Informatika setelah direvisi.

Lambang Elemen Informatika	Lambang Elemen Informatika (revisi)
 Bangunan melambangkan pondasi, pilar utama, plafon sebagai penutup sebelum berpraktik TIK. Pada semua Fase, 8 elemen dimunculkan secara eksplisit mulai Fase A s.d. F, yang dirasakan terlalu berat.	 Lingkaran adalah lambang pembelajar sepanjang hayat Pada Fase A s.d. E hanya ada 2 elemen secara eksplisit karena dilakukan pengintegrasian elemen rumah informatika; Pada Fase F elemen BK, Literasi Digital, AD, AP.

Gambar 1 Lambang Kurikulum Informatika

Penjelasan dari Lambang Elemen Informatika yang telah direvisi adalah sebagai berikut:

Lingkaran terdalam yang merupakan inti, berisi tulisan IF (singkatan dari Informatika), dan 8 (delapan) potongan *puzzle* dengan 3 (tiga) macam ukuran yang merepresentasi bobot kedalaman dari 8 (delapan) elemen lama logo rumah IF pada CP Informatika sebelum revisi, dengan rincian:

- Empat potongan *puzzle* berukuran kecil merepresentasi TIK, SK, JKI, DSI yang dikemas menjadi **Literasi digital**, adalah *core practices* warga digital di dunia nyata.
- Tiga potongan *puzzle* berukuran besar merepresentasi *core concept* informatika (BK, AD dan AP), **area pengetahuan inti** yang perlu untuk kontribusi mewujudkan dunia nyata menjadi dunia maya, konsep dasar berpikir yang diperlukan pada era digital, menuju era *Big Data* dan *Artificial Intelligence*.
- Satu potongan *puzzle* berukuran sedang mewakili PLB pada elemen lama, yang merupakan proses/koridor/jalan dari dunia nyata menuju kreator dunia maya (*engineering process*), sekaligus merekat karya nyata, melambangkan *crosscut component* dan *engineering process* yang dilakukan oleh *Computational Artefact Creator* mencipta produk dengan bergotong royong (berkolaborasi) dan berkebinekaan global.

Lingkaran tengah berwarna kuning menunjukkan tujuan dari mata pelajaran Informatika yaitu menjadi kreator dunia maya baik dalam bentuk karya digital kreatif, maupun dalam bentuk produk komputasi yang lain.

Warna kuning dalam psikologi melambangkan ceria dan semangat. Mengajak guru dan siswa untuk menikmati mata pelajaran Informatika dengan cara menyenangkan dan bersemangat. Warna kuning dalam penerapan pembelajaran yang sukses untuk abad ke-21 berdasarkan Six Thinking Hats [de Bono, Edward] melambangkan berpikir kritis.

Tantangan Bebras

(Bebras Computational Thinking Challenge)

Situs: <https://bebras.org>

Bebras Challenge (semula adalah *Algorithmic Challenge* kemudian menjadi *Computational Thinking Challenge*), diinisiasi oleh Prof. Valentina Dagiene dari Lithuania sejak tahun 2004, adalah kompetisi yang diadakan tahunan bagi siswa berumur 5 s.d. 18 tahun. Pada tahun 2019 diikuti oleh sekitar 2,944 juta siswa yang berasal dari 54 negara. Komunitas Bebras sebagian besar adalah para pembina IOI seperti halnya Indonesia, adalah sekumpulan akademisi yang peduli ke pendidikan informatika bagi siswa sekolah dasar dan menengah.

Bebras mengikuti perkembangan CT, lewat “*challenge*” atau tantangan yang diberikan untuk *problem solving* terkait informatika untuk kehidupan sehari-hari, yang disajikan secara menarik dan lucu. Lewat Tantangan Bebras, siswa diajak “membangun” keterampilan berpikir untuk menyelesaikan persoalan, yaitu melalui pendekatan *constructionism* yang diperkenalkan oleh Seymour Papert dari MIT. Siswa diajak belajar dengan mencoba menjawab tantangan. Jadi, Tantangan Bebras bukan lomba sekedar untuk menang tetapi yang lebih penting adalah untuk belajar berpikir dan menyelesaikan persoalan. Kepada peserta yang meraih peringkat tinggi, akan diberikan sertifikat.

Tujuan Tantangan Bebras:

- Memotivasi siswa agar mulai tertarik ke topik-topik informatika dan memecahkan persoalan dengan menggunakan informatika
- Menstimulasi minat siswa ke informatika
- Mendorong siswa untuk menggunakan “TIK” dengan lebih intensif dan kreatif dalam aktivitas belajarnya
- Menyemangati siswa untuk berpikir lebih dalam daripada sekedar ke komputer/alatnya dan TIK.

Tantangan bebras diselenggarakan sekali setahun pada saat hampir bersamaan di seluruh dunia, sepanjang Bebras Week, yang ditetapkan pada minggu pertama/dua bulan November.

Bebras Indonesia

Situs: <https://bebras.or.id>

Bebras dikelola oleh pembina Pusat/Nasional TOKI. Indonesia mulai bergabung ke komunitas internasional Bebras, dan untuk pertama kali mengadakan Tantangan Bebras dalam bahasa Indonesia pada tahun 2016 secara online yang terbagi dalam 3 kategori, yaitu kategori Siaga (SD/MI), kategori Penggalang (SMP/MTs), dan kategori Penegak (SMA/SMK/MA/MAK). Penyelenggaraan Tantangan Bebras dikoordinasi oleh Perguruan Tinggi yang menjadi Mitra Bebras Indonesia, dan dapat diselenggarakan di Perguruan Tinggi koordinator atau di sekolah. Peserta ada yang menggunakan komputer, tablet, bahkan handphone.

Bagaimana Berpartisipasi pada Tantangan Bebras 2019?

Pembina Bebras Indonesia bekerja sama dengan Perguruan Tinggi Mitra dengan dukungan suporter. Perguruan Tinggi (diutamakan Program Studi Informatika dan Matematika) yang berminat untuk menjadi mitra Bebras akan dihubungkan dengan Perguruan Tinggi Pembina Utama TOKI, dan sekolah yang berminat untuk mengikutsertakan siswa dapat menghubungi Perguruan Tinggi Mitra Bebras terdekat. Sebagai persiapan, Pembina Bebras tingkat Nasional juga bersedia menjadi narasumber untuk pelatihan dosen/guru yang akan bergabung.

Silahkan kontak via email ke info@bebras.or.id.

Contoh-contoh soal dan latihan online dalam Bahasa Indonesia dapat diakses di <https://olympia.id>.

Untuk Latihan di arena Bebras Indonesia

Siapun dapat berlatih secara mandiri di situs <https://latihan.bebras.or.id> dengan langkah:

1. Buka situs di url <https://latihan.bebras.or.id> selanjutnya klik tombol "Masuk" di pojok kanan atas halaman layar
2. Sign in with Google Account atau Microsoft Account
3. Kemudian buka email Anda untuk konfirmasi.
4. Klik link yang diberikan oleh Latihan Bebras ke email Anda
5. Klik "Pendaftaran" pada Course yang Anda pilih lalu klik "Daftarkan saya" pada Course yang Anda pilih

Atau dapat juga berlatih secara mandiri di situs <https://olympia.id> dengan langkah:

1. Akses "Create New Account", atau dari: <https://olympia.id/login/signup.php>
2. Setelah mengisi data diri secara lengkap dan password benar, selanjutnya klik "Create New Account"
3. Buka email Anda untuk konfirmasi.
4. Klik link yang diberikan oleh Olympia ke email Anda
5. Lakukan "Enroll" ke Bebras Challenge.

Negara-Negara Kontributor

Setiap soal di buku ini diberi bendera yang menandakan negara asal penyusun soal. Namun banyak pihak yang terlibat dalam mengedit, menerjemahkan, dan menyediakan material tambahan.

Bebras Indonesia berterima kasih kepada komunitas Bebras internasional karena memungkinkan kami untuk menggunakan soal-soal yang telah mereka kembangkan.

Bendera Negara Kontributor Soal-Soal pada Buku Kategori Penegak Tantangan Bebras 2019.



DAFTAR SOAL

Pohon Willows dan Pohon Poplars	1
Gelang Persahabatan.....	4
Rute Penerbangan	7
Petani Zacland	9
Ilmuwan di Laboratorium	12
Si Topi Merah	14
Daur Ulang Botol Gelas.....	16
Putar, Putar dan Putar	19
Pengaturan Tempat Duduk.....	21
Meniliti Rekaman	23
Membeli Sepatu	25
Semut di Sarang.....	28
Peta Yang Mana.....	30
Angka Digital.....	34
Pencacah	36

	Pohon Willows dan Pohon Poplars	PENEGAK (SMA) 2019-BE-03
---	--	-----------------------------

Bebras mempunyai sebuah aplikasi yang mampu mendapatkan halaman-halaman web bergambarkan jenis-jenis pohon tertentu berdasarkan query yang diberikan.

	
Pohon Willows	Pohon Poplars

Aplikasi itu memungkinkan pencarian yang kompleks dengan mengetikkan pertanyaan (Query):

...#...	Mengkombinasikan pertanyaan dengan dua kriteria yang mengapit simbol #. Hasilnya adalah tampilan halaman yang memenuhi kedua kriteria pertanyaan.
[...]	Me-negasi (membalikkan) pertanyaan yang ditulis di antara tanda kurung. Hasilnya adalah halaman yang BUKAN/Tidak sesuai pertanyaan.

Beberapa Contoh:

Pertanyaan / Query	Arti
willow	Tampilkan semua halaman tentang willow.
[willow]	Tampilkan halaman yang BUKAN tentang willow
willow # poplar	Tampilkan halaman yang KEDUANYA berisi informasi tentang willow DAN poplar.
[[poplar]]	Tampilkan halaman tentang poplar (= BUKAN yang BUKAN poplar).
[willow # poplar]	Tampilkan halaman yang BUKAN tentang KEDUANYA willow DAN poplar.
willow # [poplar]	Tampilkan halaman tentang willow tetapi BUKAN tentang poplars.

Tantangan: Pertanyaan mana yang akan memberikan jawaban tentang semua halaman web yang tentang willows, atau tentang Poplar.

Pilihan Jawaban:

- a) [willow] # [poplar]
- b) [[willow] # poplar] # [willow # [poplar]]
- c) [[willow] # [poplar]]
- d) [[willow] # [poplar]] # [willow # poplar]

Jawaban:

Jawaban yang benar adalah: c) [[willow] # [poplar]]

Mengapa jawaban ini yang benar? Pertama-tama mari kita lihat jawaban A yang sedikit lebih sederhana: '[willow] # [poplar]'. Query ini meminta untuk menemukan semua halaman yang memenuhi '[willow]' **dan** '[poplar]', dengan kata lain, semua halaman yang TIDAK berisi informasi tentang willow dan juga TIDAK berisi informasi tentang poplar - singkatnya: halaman yang tidak menyebutkan willow atau poplar.

Jawaban ini salah, tapi justru merupakan negasi (kebalikan) dari apa yang dicari. Negasi dinyatakan dengan '[...]'. Jadi dengan menuliskan jawaban A di antara tanda kurung, kita mendapatkan jawabannya - dan itu adalah jawaban C.

Query D dimulai dengan query C ditambah dengan persyaratan tambahan: '[willow # poplar]'. Query '[willow # poplar]' artinya bahwa halaman yang berisi informasi tentang willow dan poplar TIDAK boleh dilaporkan. Dengan kata lain: jawaban D menunjukkan apa yang dicari, kecuali halaman yang memiliki informasi tentang willow dan poplar. Jadi, query D akan mengarah ke halaman yang memiliki informasi tentang willow atau poplar, tetapi tidak keduanya.

Query D dimulai dengan query C tetapi menambahkan persyaratan tambahan: '[willow # poplar]'. Permintaan '[willow # poplar]' menunjukkan bahwa halaman yang berisi informasi tentang kedua jenis pohon willow dan poplar seharusnya TIDAK dilaporkan. Dengan kata lain: jawaban D menampilkan apa yang Bebras cari, kecuali halaman-halaman yang memiliki informasi tentang kedua jenis pohon willow dan poplar. Jadi, permintaan D akan menghasilkan halaman-halaman yang berisi informasi tentang entah willow atau poplar, tetapi tidak keduanya.

Untuk memahami apa yang dicari oleh query B, kita membaginya menjadi dua bagian. Bagian pertama '[willow] # poplar]' menunjukkan bahwa kita tidak menginginkan halaman-halaman yang tentang poplar dan bukan tentang willow, dengan kata lain, jika halaman tersebut tentang poplar, maka halaman tersebut juga harus tentang willow. Begitu pula, bagian kedua '[willow # [poplar]]' menyatakan bahwa jika halaman tersebut berisi informasi tentang willow, maka halaman tersebut juga harus berisi informasi tentang poplar. Menggabungkan kedua persyaratan (= query B) akan menghasilkan halaman-halaman yang harus berisi tentang kedua willow dan poplar atau tidak mengandung informasi tentang keduanya sama sekali.

Ini Informatika!

Seperti aplikasi pencarian imajiner, mesin pencari modern kita memungkinkan permintaan untuk digabungkan dalam beberapa cara - menggunakan NOT, AND, atau OR. Apa yang sebenarnya diinginkan oleh Bebras adalah mencari willow ATAU poplar, tetapi sayangnya tidak ada simbol 'OR' dalam aplikasi pencarian miliknya.

Tugas ini menunjukkan bahwa 'OR' dapat diungkapkan dengan menggunakan kombinasi 'NOT' dan 'AND', khususnya

$a \text{ OR } b = \text{NOT}(\text{NOT}(a) \text{ AND } \text{NOT}(b))$

Logika mengetahui ini sebagai salah satu hukum De Morgan.

Anda menggunakan banyak ungkapan logika saat pemrograman. Mengetahui beberapa hukum dasar seperti hukum De Morgan membuat lebih mudah untuk menulis dan memahami program-program kompleks.

Kita harus berhati-hati tentang kata 'OR' karena dapat memiliki arti yang berbeda. Salah satu jawaban yang salah - perintah D - menghasilkan apa yang kita sebut sebagai OR eksklusif (atau XOR): situs web tentang willow atau poplar, tetapi tidak keduanya. Ini adalah OR yang sering kita gunakan dalam kehidupan sehari-hari. Jika seseorang bertanya apakah Anda ingin makan kentang goreng atau pizza, biasanya Anda tidak menjawab bahwa Anda ingin makan keduanya.

Perintah C menghasilkan OR inklusif: halaman tentang willow atau tentang poplar atau tentang keduanya.

Catatan terakhir: Anda mungkin merasa aneh bahwa aplikasi pencarian menggunakan simbol '#' untuk AND - mungkin Anda mengharapkan kata 'AND' atau simbol '&' - dan tanda kurung siku untuk NOT. Bagi komputer ini tidak masalah, komputer hanya melakukan apa yang diprogram untuk dilakukan. Biasanya, para programmer akan memilih konvensi yang mudah dipahami manusia - untuk menghindari kebingungan - tetapi komputer tidak peduli sama sekali: bagi komputer, '#' sama seperti simbol lainnya.

Authorship

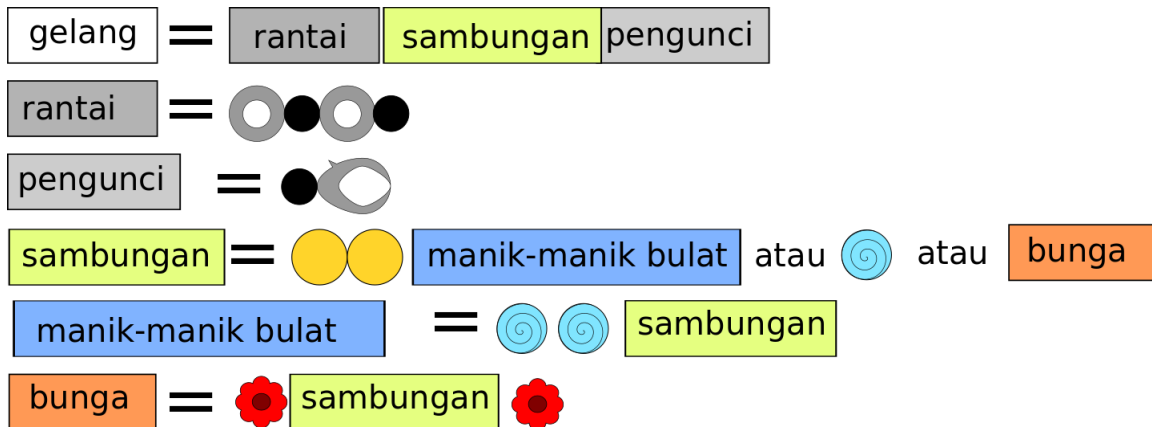
- 2019-03-25 Kris Coolsaet kris.coolsaet@ugent.be (corresponding author)
 - Arne Goeteyn arne.goeteyn@ugent.be
 - Robin Deklerck robin.deklerck@ugent.be
- Task Proposal

License

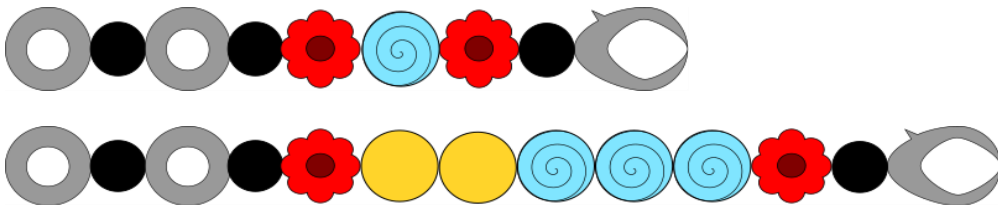
Copyright © 2019 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

	Gelang Persahabatan	PENEGAK (SMA) 2019-BE-04
---	----------------------------	-----------------------------

Santi membuat gelang dimulai dengan gelang dan merangkainya sesuai dengan aturan sebagai berikut:

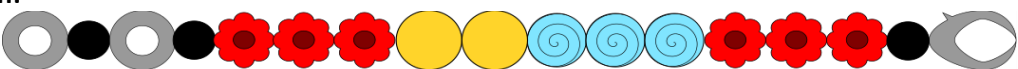
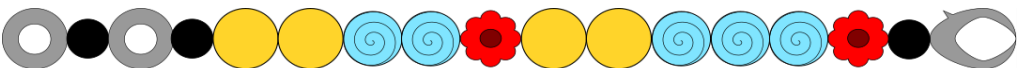
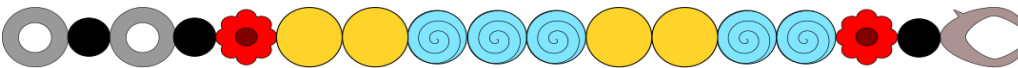
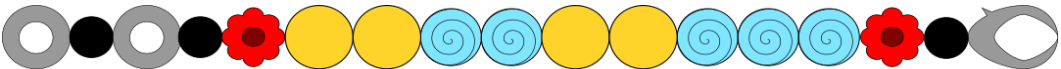


Artinya, setiap simbol di sebelah kiri diganti dengan salah satu urutan simbol yang ditunjukkannya. Misalnya, dengan menggunakan aturan ini beberapa kali, Santi dapat membuat dua gelang sebagai berikut.

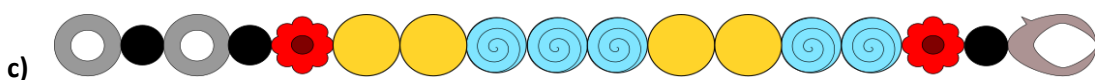


Tantangan: Dari beberapa gelang berikut ini, gelang mana yang tidak mengikuti aturan?

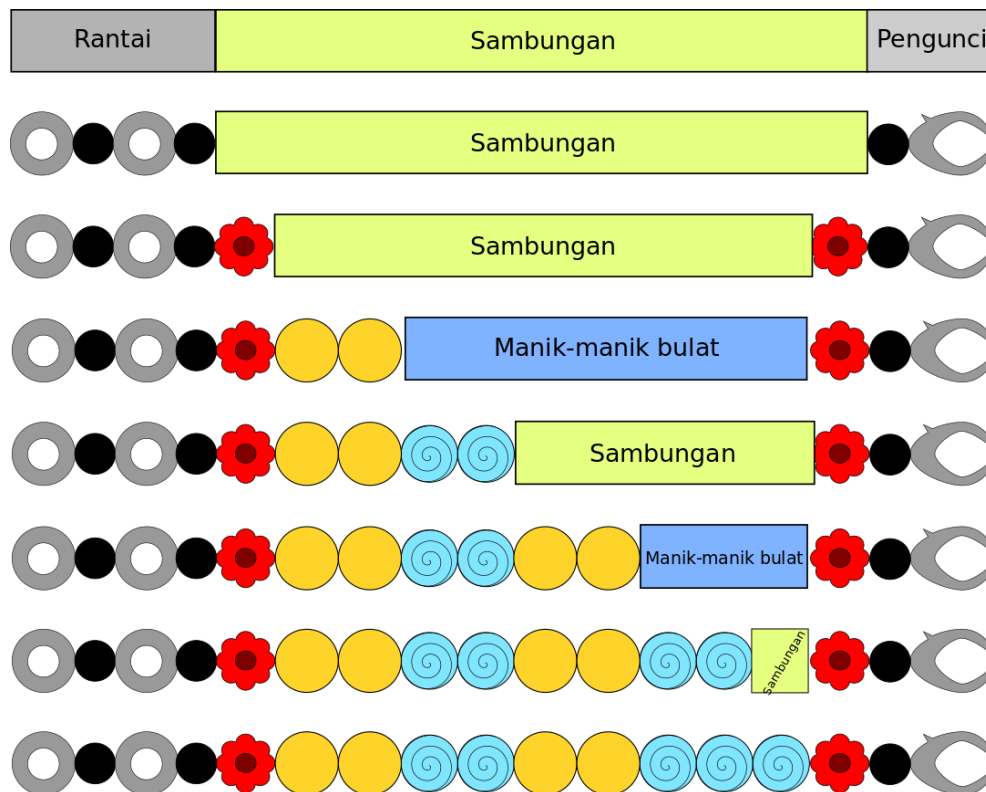
Jawaban:

- a) 
- b) 
- c) 
- d) 

Jawaban yang benar adalah:



Dari pada membuktikan bahwa c) memiliki kesalahan, mungkin lebih mudah untuk menunjukkan bahwa tiga gelang lainnya memang dapat dibuat menggunakan aturan-aturan ini. Untuk gelang d), kami menunjukkannya dalam gambar di bawah ini:



Tidak begitu sulit untuk membuat gambar serupa untuk a) dan b).

Untuk membuktikan bahwa c) tidak mungkin dibuat menggunakan aturan-aturan tersebut, kita melihat aturan 'Sambungan', aturan 'Manik-manik bulat', dan aturan 'bunga'.

Ketiga aturan ini dapat digabungkan menjadi aturan baru, yang tidak mengubah jenis gelang yang dapat kita buat:

Dengan kata lain, jika kita mengabaikan manik bunga, maka setiap gelang terdiri dari pola yang diulang beberapa kali (mungkin hanya sekali, atau bahkan tidak sama sekali), diikuti oleh sebuah manik biru. Gelang a), b), dan d) jelas mengikuti pola ini, tetapi gelang c) tidak - ada tiga sambungan berwarna biru muda yang berbentuk 'swirly' di tengah gelang, dan itu tidak diizinkan.

Ini Informatika!

Kumpulan aturan yang diberikan dalam tugas ini dapat digunakan oleh komputer untuk memeriksa apakah sebuah gelang dapat dibuat oleh Santi. Aturan-aturan serupa digunakan oleh komputer untuk memeriksa apakah para programmer membuat kesalahan pengetikan dalam program-program mereka, atau untuk memeriksa apakah apa yang Anda ketikkan ke dalam formulir web benar. Sebagai contoh, aturan-aturan berikut:

- number → unsigned or + unsigned or - unsigned
- unsigned → digit or unsigned digit
- digit → 0 or 1 or 2 or 3 or 4 or 5 or 6 or 7 or 8 or 9

dapat digunakan untuk memberi tahu komputer apa itu angka yang valid.

Tipe-tipe kumpulan aturan seperti ini, dalam ilmu komputer, disebut sebagai tata bahasa atau khususnya tata bahasa bebas konteks.

Authorship

- 2019-03-25 Ellen Vanhove (Belgium), ellen.vanhove@ugent.be &
- 2019-03-26 Kris Coolsaet (Belgium), kris.coolsaet@ugent.be (corresponding author)

Task Proposal

- 2019-05-14 Bundit Thanasopon (Thailand), bundit@it.kmitl.ac.th; Mile Jovanov (Macedonia), mile.jovanov@gmail.com; J. P. Pretti (Canada), jpretti@uwaterloo.ca:
- Edit wording and graphics.

License

Copyright © 2019 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

	Rute Penerbangan	PENEGAK (SMA) 2019-BE-07
---	---	-----------------------------

Perusahaan penerbangan Bebras International memiliki rute penerbangan yang menghubungkan beberapa kota di dunia seperti ditunjukkan pada gambar sebagai berikut:



Emisi gas CO₂ adalah penyebab utama pemanasan global. Untuk mengurangi CO₂, perusahaan berencana untuk menghapus rute, yang tetap memungkinkan pelanggan pasti bisa bepergian dari satu kota ke kota lainnya. Misalnya, jika rute dari San Francisco ke Washington D.C dihapus, pelanggan masih dapat bepergian dari San Francisco ke New York, kemudian dari New York ke Washington D.C.

Tantangan : Untuk rute yang ditunjukkan di atas, berapa (maksimal) rute yang dapat dihapus?

Pilihan Jawaban:

- a) 6
- b) 7
- c) 8
- d) 9

Jawaban yang tepat: c) 8.

Contoh di bawah ini menunjukkan bahwa memungkinkan untuk menghapus 8 rute dan masih memungkinkan pelanggan terbang ke kota mana pun.

Perhatikan bahwa ketika kita menghapus 8 rute, kita akan memiliki $15 - 8 = 7$ rute yang tersisa. Mengapa kita tidak bisa menghapus lebih banyak rute? Jika hanya ada 2 kota, jelas bahwa kita memerlukan 1 rute. Jika ada 3 kota, maka kita memerlukan 2 rute. Pola ini terus berlanjut dan ketika ada 8 kota, kita memerlukan 7 rute.

Untuk memastikan bahwa pola ini benar, kita perlu meyakinkan diri sendiri bahwa jika Anda menghapus 9 rute (atau lebih), maka pelanggan tidak selalu dapat terbang ke kota mana pun. Jika 9 rute (atau lebih) dihapus, maka akan ada hanya 6 rute (atau kurang) yang tersisa. Apa yang salah jika hanya ada 6 rute? Jika hanya ada 6 rute, karena ada 8 kota, maka entah ada kota yang tidak berada di

rute mana pun atau ada kota yang hanya berada di satu rute. Jika ada kota yang hanya berada di satu rute, maka kita bisa menghapus kota dan rute ini dan tersisa 5 rute dan 7 kota. Melanjutkan argumen ini, dalam semua kasus, kita akan tersisa dengan kota yang tidak berada di rute atau akhirnya tidak ada rute antara 2 kota. Oleh karena itu, jika hanya ada 6 rute, maka pelanggan tidak selalu dapat terbang ke kota mana pun.

Ini Informatika!

Rute penerbangan dapat digambarkan sebagai graf. Kota adalah simpul dan rute adalah jalur. Masalah untuk mengurangi sebanyak mungkin jumlah rute, sambil memastikan bahwa semua kota tetap terhubung, adalah aplikasi dari masalah Minimum Spanning Tree (MST) dalam Ilmu Komputer. Masalah MST adalah mengekstraksi pohon yang menghubungkan semua simpul dari suatu graf. Untuk informasi lebih lanjut, bacalah https://en.wikipedia.org/wiki/Minimum_spanning_tree. Informatikawan menggunakan algoritma MST untuk menyelesaikan persoalan-persoalan jaringan telekomunikasi, transportasi atau penyaluran air.

Authorship

- 2019-04-07 Sébastien Combéfis (Belgium), sebastien@combefis.be
- Task Proposal
- 2019-05-14 Eslam AbdElAal (Egypt) (eslam@aast.edu), Georgios Fesakis (Greece) (gfesakis@aegean.gr), J. P. Pretti (Canada), jpretti@uwaterloo.ca, Vipul Shah (India), v.shah@tcs.com

License

Copyright © 2019 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

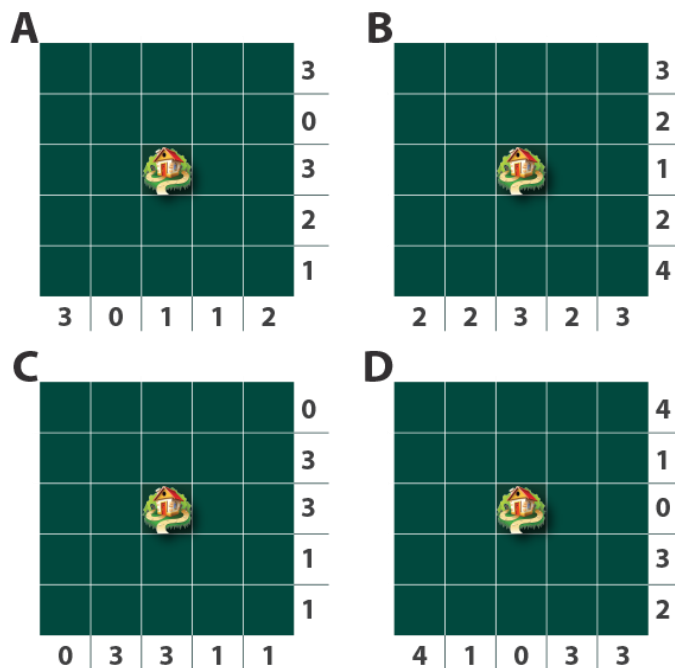
	Petani Zacland	PENEGAK (SMA) 2019-CY-02
---	-------------------------	-----------------------------

Lahan pertanian di Zacland dibagi menjadi petak-petak. Setiap tahun, para petani harus memutuskan apakah tanah pertaniannya akan dipakai menanam gandum (warna kuning), atau rumput (hijau), dan mereka harus melaporkan ladang gandumnya ke pemerintah. Lapornya harus berisi banyaknya petak ditanami gandum pada setiap baris dan kolom, yang akan dibaca melalui satelit. Salah satu laporan petani diberikan sebagai berikut:

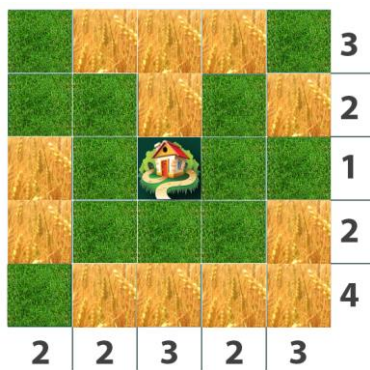


Tantangan: Yang mana di antara gambar sebagai berikut, yang akan menghasilkan laporan yang akurat?

Pilihan Jawaban:



Jawaban yang tepat adalah: B.



Mengapa jawaban-jawaban lainnya salah?

Pada jawaban A dan D, jumlah bidang dengan gandum di baris dan kolom tidak sesuai:

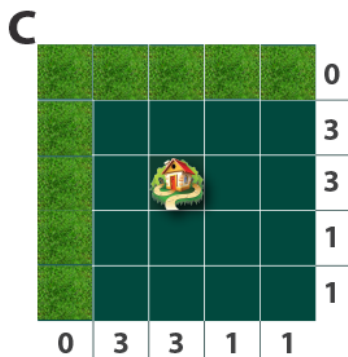
A: baris: 7, kolom: 9

D: baris: 11, kolom: 10.

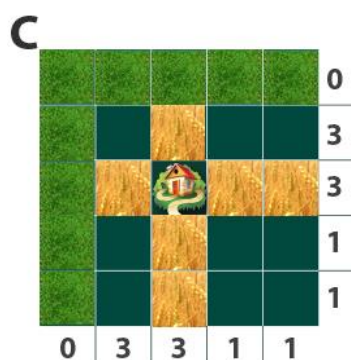
Karena jumlah baris dan kolom pada jawaban C sesuai, kita harus berargumen dengan cara berbeda:

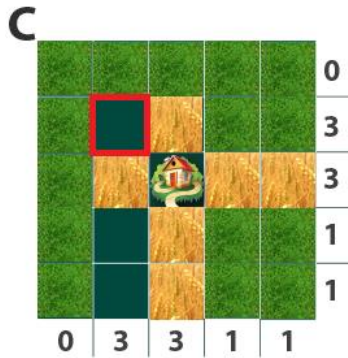
Baris pertama dan kolom pertama harus kosong:

Hanya ada satu cara untuk menutupi bidang-bidang di baris ketiga dan kolom ketiga dengan gandum:



Kemudian, semua bidang lainnya dalam kolom 4 dan 5 harus ditutupi oleh rumput.





Setelah itu, tidak mungkin menanam gandum di dua bidang lainnya di baris kedua karena hanya ada satu bidang lagi yang tersedia (ditandai dengan warna merah).

Jadi, jawaban C salah.

Ini Informatika!

Array adalah struktur data fundamental dalam pemrograman. Sebuah array terdiri dari objek-objek serupa, dalam kasus kita adalah batang pohon, yang diatur dalam baris dan kolom. Selain itu, jumlah dari suatu baris atau jumlah dari suatu kolom juga merupakan hal yang fundamental dalam pemrograman awal.

Algoritma Backtracking: Masalah ini dapat dianggap dapat dipecahkan dengan menggunakan konsep dasar dari algoritma Backtracking. Solusi dibangun secara bertahap, satu bagian pada satu waktu, dengan menghilangkan solusi yang gagal memenuhi batasan masalah.

Untuk informasi lebih lanjut, lihat <https://en.wikipedia.org/wiki/Backtracking>.

Authorship

- 2019-03-11 Thomas Ioannou (Cyprus), ioannouthomas@gmail.com: Task Proposal

License

Copyright © 2019 Bebras – International Contest on Informatics and Computational Thinking. This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <https://creativecommons.org/licenses/by-sa/4.0/>

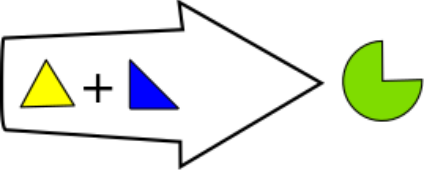
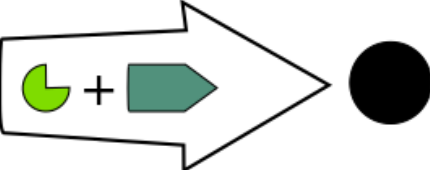
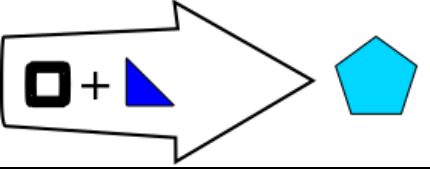
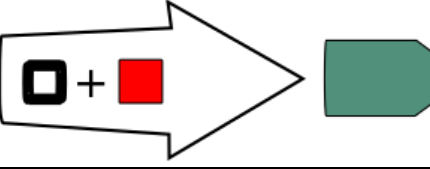
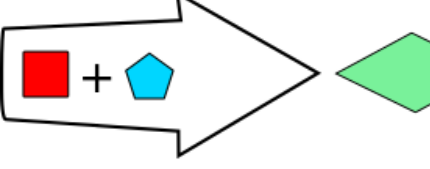
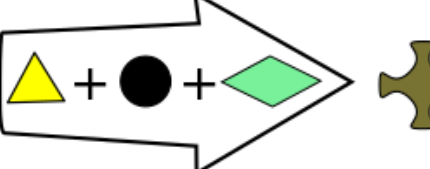
	Ilmuwan di Laboratorium	PENEGAK (SMA) 2019-HU-04
---	--------------------------------	-----------------------------

Tiga orang Ilmuwan Charli (C), Paula (P), dan Bary (B), mula-mula hanya mempunyai 4 (empat) jenis

bahan kimia    

Mereka ingin membentuk sebuah bahan kimia baru, yaitu 

Setiap ilmuwan bekerja selama dua hari. Setiap harinya, hanya satu ilmuwan yang bekerja dan hanya bisa menghasilkan salah satu proses saja. Satu proses artinya adalah mencampurkan dua zat kimia untuk membentuk zat kimia baru, yang bisa dipakai untuk proses berikutnya. Setiap ilmuwan hanya dapat melakukan dua proses seperti ditunjukkan pada gambar:

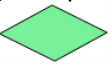
Ahli Kimia Charli (C)		
Ahli Fisika Paula (P)		
Ahli Biologi Bary (B)		

Tantangan: Urutan huruf inisial Ilmuwan mana yang menunjukkan urutan mereka bekerja, yang tidak memungkinkan menghasilkan produk akhir 

Pilihan Jawaban:

- a) C P B P C B
- b) P P C C B B
- c) C P B C P B
- d) P B P C C B

Jawaban yang tepat adalah: c) C P B C P B

Karena ini adalah proses keduanya yang menciptakan produk akhir, B (Ahli Biologi Bary) harus menjadi orang terakhir dalam urutan. Agar dapat menciptakan produk akhir menggunakannya yang kedua, dia memerlukan:  

Untuk menghasilkan lingkaran hitam , C (Ahli Kimia Carli) perlu memiliki akses ke  dan harus pergi ke laboratorium setelah P (Ahli Fisika Paula). C juga harus melakukan operasi pertamanya

sebelum operasi keduanya. Untuk melakukan operasi pertamanya, B (Ahli Biologi Bary) membutuhkan P untuk melakukan operasi pertamanya.

Pada jawaban C, P pertama dalam urutan hanya dapat menciptakan salah satu bahan kimia yang diperlukan untuk melakukan dua operasi berikutnya. Salah satunya pasti gagal. Urutan C P B C P B tidak akan berhasil menciptakan produk akhir.

Ini Informatika!

Dalam Ilmu Komputer dan pemrograman, kita memiliki proses di mana kita menggunakan masukan dan menciptakan keluaran. Penting untuk dipahami bahwa kita tidak dapat memulai sebuah proses jika masukan yang dibutuhkan, yang diperoleh dari keluaran proses sebelumnya, tidak tersedia. Programmer harus mengorganisir tugas-tugas kecil untuk memastikan keluaran yang dibutuhkan dari proses sebelumnya sudah tersedia.

Seorang programmer yang baik harus mampu mengantrikan proses dengan tepat.

Authorship

List all authors who have contributed to this Document, with full name, e-mail and country.

- András Béni - HU, pluharzs@caesar.elte.hu, 2019-03-26
- Zsuzsa Pluhár- HU, pluharzs@caesar.elte.hu, 2019-03-26 (only pictures)
- Graeme Buckie – AU, graeme.buckie@csiro.au 2019-05-16

License

Copyright © 2019 Bebras – International Contest on Informatics and Computer Fluency.

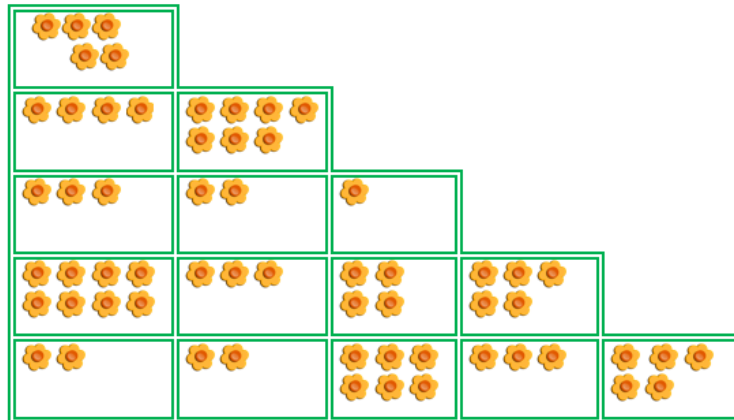
This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License. <http://creativecommons.org/licenses/by-nc-sa/3.0/>.



Si Topi Merah

PENEBAK (SMA)
2019-RO-02

Si Topi Merah ingin memungut bunga di kebun nenek. Kebun tersebut dibagi menjadi beberapa bagian, setiap bagian mempunyai sejumlah bunga. Topi Merah mulai dari kiri atas dan berjalan hanya dengan arah turun atau ke kanan, sampai ke pojok kanan bawah.



Tantangan: Berapa banyaknya bunga yang paling banyak dapat dikumpulkan oleh si Topi Merah?

Pilihan Jawaban:

- a) 35
- b) 38
- c) 58
- d) 41

Jawaban:

Jawaban yang tepat adalah: d) 41.

Pada setiap bagian, kita dapat menghitung jumlah maksimum bunga yang dapat dikumpulkan oleh Red Riding Hood dari awal hingga bagian tersebut (mari sebut ini sebagai maksimum). Maksimum dapat dihitung sebagai berikut:

- Jika suatu bagian hanya dapat dijangkau dari satu bagian sebelumnya, maka maksimumnya adalah jumlah bunga di bagian tersebut ditambah dengan maksimum dari bagian sebelumnya.
- Jika suatu bagian dapat dijangkau dari dua bagian sebelumnya, maka maksimumnya adalah jumlah bunga di bagian tersebut ditambah dengan yang lebih besar dari kedua bagian sebelumnya.

Ini Informatika!

Solusi ini adalah penerapan strategi "dynamic programming" untuk memperoleh solusi yang optimal. Prinsip pemrograman dinamis adalah bahwa ada hasil-hasil perhitungan dari sub masalah yang dihitung terlebih dahulu. Dalam kasus kita, itu adalah jumlah maksimum bunga yang dikumpulkan dari titik awal ke salah satu dari bagian-bagian tengah. Triknya adalah beberapa hasil perhitungan submasalah yang sudah dihitung digunakan untuk menghitung hasil perhitungan submasalah yang baru. Dengan menerapkan ide ini, gelombang banjir (*flood wave*) hasil perhitungan submasalah yang sudah dihitung berasal dari titik awal. Ketika mencapai titik akhir, maka kita memiliki hasil akhir yang

diinginkan. Banyak masalah optimisasi praktis dapat dipecahkan dengan strategi ini, yang sering menghasilkan algoritma yang sangat efisien dalam hal waktu.

https://en.wikipedia.org/wiki/Dynamic_programming

Authorship

- 2019-03-29 Laura Ungureanu (Romania), lauungureanu@gmail.com

License

Copyright © 2019 Bebras – International Challenge on Informatics and Computational Thinking.

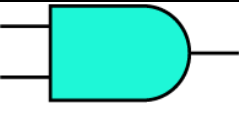
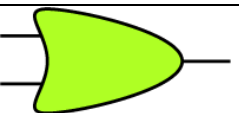
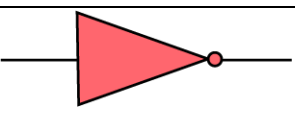
This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).



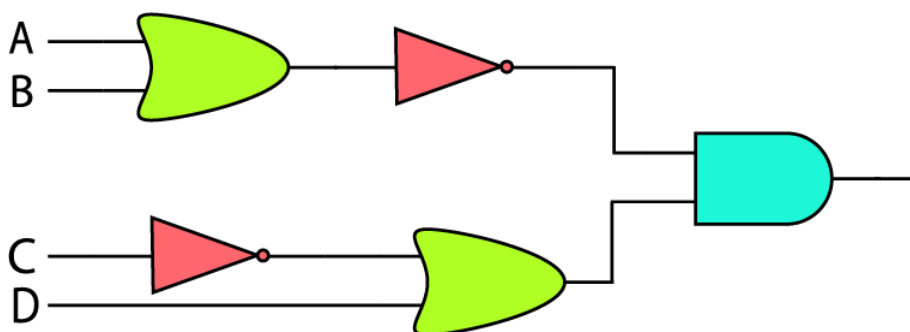
Daur Ulang Botol Gelas

PENEGAK (SMA)
2019-RO-03

Ibu Kepala Sekolah Bebras sangat sibuk berurusan dengan botol gelas di sekolah. Ia mengamati bahwa gelas di sekolahnya berwarna putih atau warna warni lainnya. Untungnya, ia punya 3 buah mesin yang dapat mengubah gelas. Dua buah mesinnya menerima dua jenis gelas dan mengubahnya. Satu mesin lainnya menerima sebuah gelas dan mengubah warnanya.

	Mesin ini akan menghasilkan gelas putih jika dua gelas putih dimasukkan. Kombinasi warna lain akan menghasilkan gelas berwarna
	Mesin ini akan menghasilkan gelas berwarna jika dua buah gelas berwarna dimasukkan. Kombinasi lainnya akan menghasilkan gelas putih
	Mesin ini akan mengubah gelas berwarna menjadi putih, atau mengubah gelas putih menjadi berwarna

Ia merangkai mesin menjadi sebuah sistem sebagai berikut:



Tantangan:

Gelas apa yang harus dimasukkan ke A, B, C, D sehingga hasilnya adalah gelas putih?

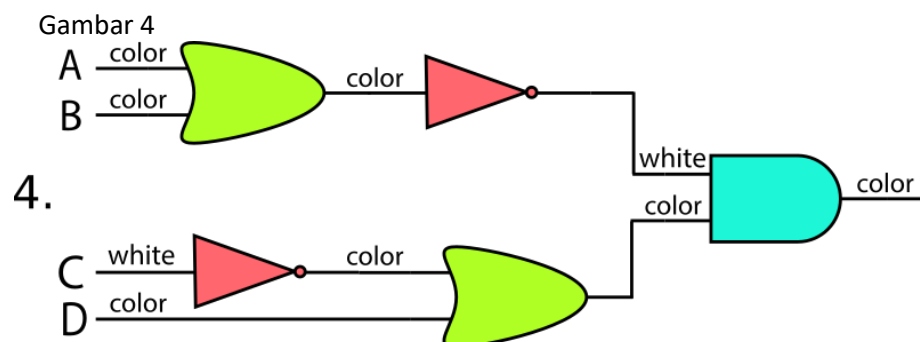
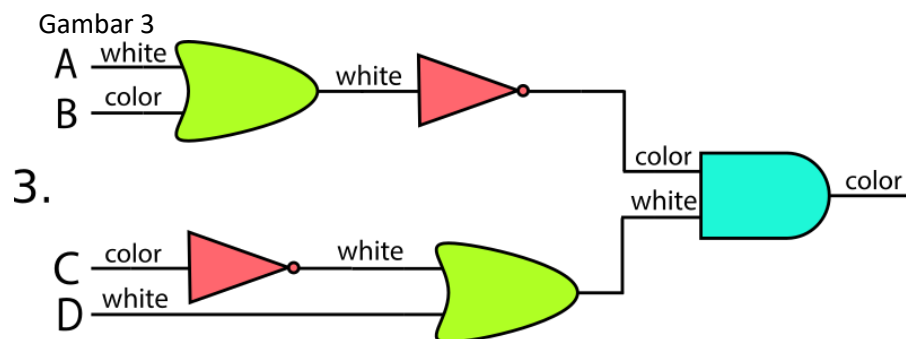
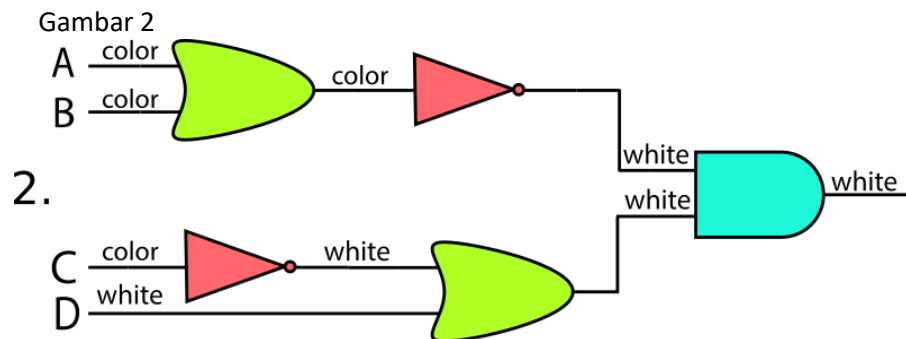
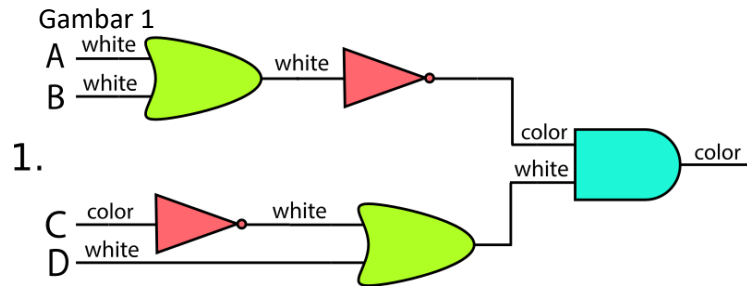
Pilihan Jawaban:

- a) A = berwarna, B = berwarna, C = berwarna, D = putih
- b) A = putih, B = putih, C = berwarna, D = putih
- c) A = putih, B = berwarna, C = berwarna, D = putih
- d) A = berwarna, B = berwarna, C = putih, D = berwarna

Jawaban:

Jawaban yang tepat adalah **a)** A = berwarna, B = berwarna, C = berwarna, D = putih

Kita dapat menggambar gambar untuk setiap pilihan dan menuliskan jenis gelas di atas setiap garis (gelas yang dimasukkan dan yang dihasilkan). Dengan cara ini, kita dapat memeriksa hasil keluaran dari setiap pilihan dan menemukan yang benar. Satu-satunya gambar dari pilihan-pilihan ini dengan gelas putih sebagai hasil keluaran adalah: Gambar 2.



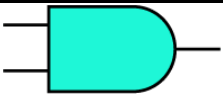
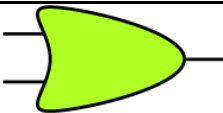
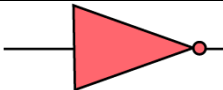
Perhatikan bahwa ada tiga masukan yang mungkin menghasilkan gelas putih sebagai keluaran:

- A = berwarna, B = berwarna, C = berwarna, D = berwarna
- A = berwarna, B = berwarna, C = berwarna, D = putih (ini adalah jawaban a) di atas)
- A = berwarna, B = berwarna, C = putih, D = putih

Ini Informatika!

Semua komputer memiliki rangkaian, yang terdiri dari berbagai jenis elemen kecil yang disebut gerbang. Beberapa gerbang yang paling populer adalah NOT, OR, AND, dan XOR.

Di sini kita memiliki gerbang AND, OR, dan NOT (representasi grafis mereka dalam tugas ini sama seperti yang digunakan dalam Teknik Elektro). Dalam Teknik Elektro, elemen-elemen tersebut bekerja menggunakan sinyal-sinyal listrik. Kami menyatakan mereka sebagai 1 jika ada sinyal, dan 0 jika tidak ada sinyal. Dalam logika, kami merujuk pada sinyal 1 sebagai BENAR (TRUE) dan pada sinyal 0 sebagai SALAH (FALSE). Dalam tugas ini, kami mempertimbangkan kaca putih sebagai BENAR (atau 1) dan kaca berwarna sebagai SALAH (atau 0).

	Gerbang AND membutuhkan kedua input menjadi BENAR (gelas putih) untuk mendapatkan BENAR sebagai keluaran.
	Gerbang OR membutuhkan setidaknya salah satu dari input menjadi BENAR untuk mendapatkan BENAR sebagai keluaran.
	Gerbang NOT membalik nilai, jadi jika inputnya BENAR , maka keluarannya SALAH , dan jika inputnya SALAH , maka keluarannya BENAR .

Gerbang logika digunakan, antara lain, dalam arsitektur mikroprosesor untuk melakukan operasi aritmatika dan logika.

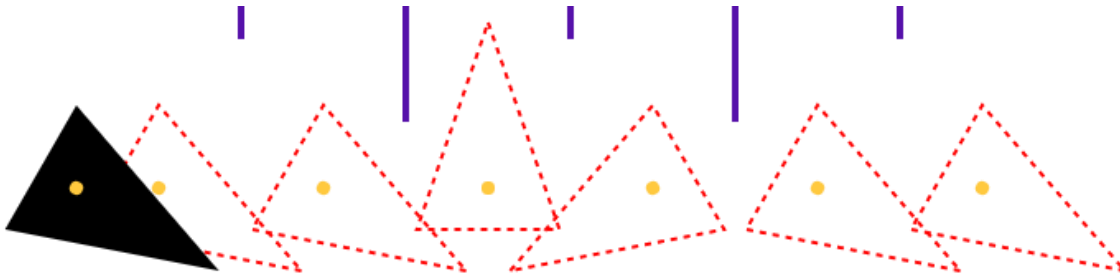
https://en.wikipedia.org/wiki/Logic_gate

Authorship

- 2019-03-29 Laura Ungureanu (Romania), lauungureanu@gmail.com Task Proposal
- 2019-03-29 Corina Vint (Romania), corina.vint@yahoo.com
- 2019-05-14 Arnheidur Gudmundsdottir (Iceland) arnheidur@sky.is and Margot Phillipps (New Zealand) margot.phillipps@gmail.com. Text changes in the workshop.
- 2019-05-15 WG4: Alisher Ikramov (Uzbekistan), ikramov.alisher@list.ru; Bohdan Kudrenko (Ukraine), kudrenkobo@gmail.com

License

Copyright © 2019 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).



Sebuah segitiga bergerak di atas ban berjalan. Segitiga itu mempunyai titik (poros) putar yang ditandai bulatan kecil. Tiga titik sudut segitiga, hanya satu yang posisinya lebih jauh ke poros. Jika segitiga berputar 120 derajat, salah satu titik sudut segitiga posisinya selalu di atas poros.

Ada sejumlah batang di atas ban berjalan. Ada dua jenis batang:

- Batang yang panjang, selalu memutar segitiga sebesar 120 derajat berlawanan arah jarum jam
- Batang yang pendek hanya akan memutar segitiga jika titik sudut segitiga yang terjauh dari titik poros sedang berada di atas

Bob merancang proses yang menggunakan segitiga dan batang seperti yang ditunjukkan di atas. Dia ingin agar segitiga yang masuk ban berjalan dalam berbagai orientasi untuk diatur menjadi orientasi yang sama di akhir ban berjalan.

Bob ingin Anda menemukan urutan batang yang mampu membuat segitiga yang awalnya dengan orientasi apa pun, akan berputar dan berakhir dengan orientasi yang selalu sama di akhir ban berjalan. Bob tidak peduli dengan orientasi akhir segitiga, asalkan segitiga selalu berorientasi sama.

Tantangan: Urutan yang mana akan menghasilkan sesuai harapan Bob?

Pilihan Jawaban:

- a) Pendek, Panjang, Pendek, Panjang
- b) Panjang, Pendek, Panjang, Pendek
- c) Pendek, Panjang, Panjang, Pendek
- d) Panjang, Pendek, Pendek, Panjang

Jawaban:

Jawaban yang tepat adalah **c)** Pendek, Panjang, Panjang, Pendek

Untuk memeriksa bahwa posisi pendek, panjang, panjang, posisi pendek mengarah pada rotasi akhir yang sama, mari kita periksa semua rotasi awal: Semua variasi lain mengarah pada rotasi akhir yang berbeda.

Ini Informatika!

Tugas ini mengenai mesin berkeadaan terbatas (*finite-state machine*) seperti yang ditunjukkan di sini: Mesin ini memiliki tiga keadaan, di mana titik paling jauh dari poros adalah: ATAS (UP), KIRI (LEFT), atau KANAN (RIGHT).

Ada dua operator transisi: tangan pendek (*shorthand*) dan tangan panjang (*long hand*).

Dari setiap keadaan, segitiga dapat melompat ke keadaan lain melalui sebuah transisi.

Dalam tugas ini, kita harus menemukan urutan transisi yang selalu menghasilkan keadaan akhir yang sama, tidak peduli keadaan awalnya. Jadi, kita diminta untuk menghasilkan sebuah urutan reset. Ini adalah tugas yang cukup umum dalam Ilmu Komputer.

Di bawah ini adalah tautan ke situasi serupa yang melibatkan teks daripada segitiga!
[Synchronizing words](#) dalam finite automata.

Authorship

- 2019-04-01 Ilya Posov (Russia), iposov@gmail.com:
Task Proposal
- 2019-05-14 Chris Roffey (UK), chris@codingclub.co.uk; Jana Tomcsanyiova (Slovakia), j.tomcsanyiova@gmail.com; Agnes Erdosne Nemeth (Hungary), agi@microprof.hu:
Edits at conference (2018-CA-05)

License

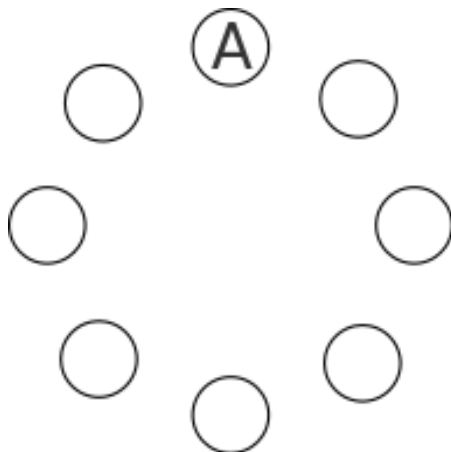
Copyright © 2019 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#).



Pengaturan Tempat Duduk

PENEGAK (SMA)
2019-CA-05

8 orang sahabat (Ani, Buni, Cici, Dani, Eni, Fani, Gani, Hani) duduk melingkar mengelilingi sebuah meja bundar seperti ditunjukkan pada gambar sebagai berikut:



Diketahui beberapa fakta sebagai berikut:

- Ani duduk di seberang (berlawanan) dengan Dani.
- Hani duduk di antara Gani dan Eni.
- Fani tidak disebelah Ani maupun Dani.
- Ada satu orang di antara Gani dan Cici.
- Eni di sebelah kiri Dani.

Tantangan: Tentukan urutan tempat duduk ke 8 sahabat tersebut, dengan menuliskan huruf-huruf awal nama mereka saja, searah jarum jam, dimulai dari A. Contoh jawaban: AHCFGBED

Jawaban:

Jawaban yang benar adalah **ACFBDEHG**.

Fakta 1, Ani duduk tepat di hadapan Dani, memungkinkan kita menempatkan Dani.

Sekarang, Fakta 5, Eni berada di samping Dani, di sebelah kiri Dani, memungkinkan kita menempatkan Eni.

Pada tahap ini, Fakta 2, Hani duduk di antara Gani dan Eni, memberitahu kita di mana Hani duduk. Mengetahui di mana Hani duduk, kita sekarang dapat menempatkan Gani.

Dengan lima teman ditempatkan, Fakta 3, Fani tidak berada di samping Ani atau Dani, hanya meninggalkan satu tempat untuk Fani.

Fakta 4, Ada satu orang di antara Gani dan Cici, sekarang memberitahu kita di mana Cici duduk.

Akhirnya, hanya ada satu kursi dan satu teman yang harus ditempatkan, jadi kita dapat menempatkan Buni.

Ini Informatika!

Logika adalah tentang mengikuti aturan, memahami urutan, dan negasi.

Sebagai contoh, operator logika kunci yang digunakan dalam pertanyaan ini adalah *negasi*, yang digunakan setiap kali kita mengatakan "tidak". Jadi, fakta yang menyatakan bahwa "Fani tidak berada di samping Ani atau Dani" dapat dinyatakan secara berbeda sebagai "Fani tidak berada di samping (Ani atau Dani)" yang sama dengan "(Fani tidak berada di samping Ani) dan (Fani tidak berada di samping Dani)".

Para siswa yang lebih tua mungkin tertarik dengan fakta bahwa operator negasi, "not", mengubah ekspresi logika dari not (A atau B) menjadi (not A) dan (not B), yang merupakan contoh hukum De Morgan.

Mengetahui bagaimana ekspresi logika dapat ditulis ulang, digabungkan, atau disederhanakan adalah keterampilan yang sangat berguna dalam ilmu komputer.

Authorship

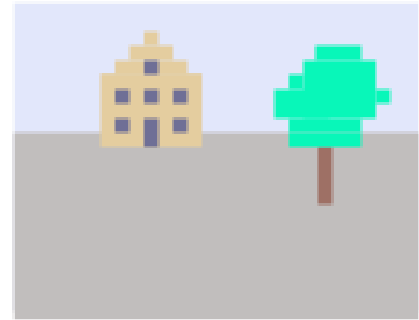
- 2019-04-01 Troy Vasiga (Canada), troy.vasiga@uwaterloo.ca:
Task Proposal
- 2019-05-14 Chris Roffey (UK), chris@codingclub.co.uk; Jana Tomcsanyiova (Slovakia), j.tomcsanyiova@gmail.com; Agnes Erdosne Nemeth (Hungary), agi@microprof.hu

License

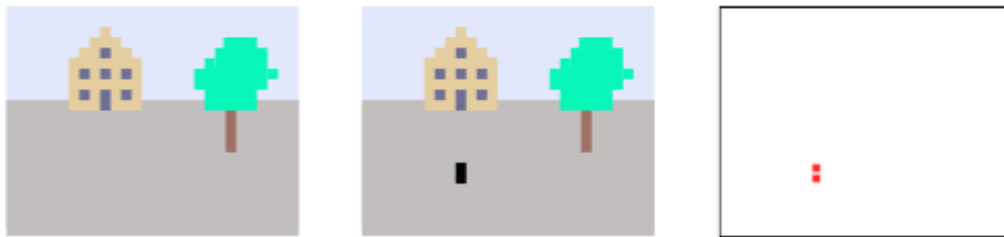
Copyright © 2019 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

	Meneliti Rekaman	PENEGAK (SMA) 2019-DE-03
---	-------------------------	-----------------------------

Setiap 10 detik, sebuah kamera merekam foto sebuah lapangan di depan gedung. Sebuah program komputer membandingkan sebuah foto dengan foto sebelumnya, dan membuat gambar yang menyatakan perbedaannya. Pada gambar perbedaan ini, ada kotak merah pada setiap posisi di mana foto tsb tampak berbeda dengan sebelumnya. Jika perbedaan putih artinya tidak ada bedanya.



Misalnya, untuk kedua foto sebagai berikut, perbedaan gambar ditunjukkan sebagai berikut:



Berikut ini anda lihat sederet gambar yang diambil selama 50 detik. 5 gambar terekam pada periode tersebut.

Kejadian:

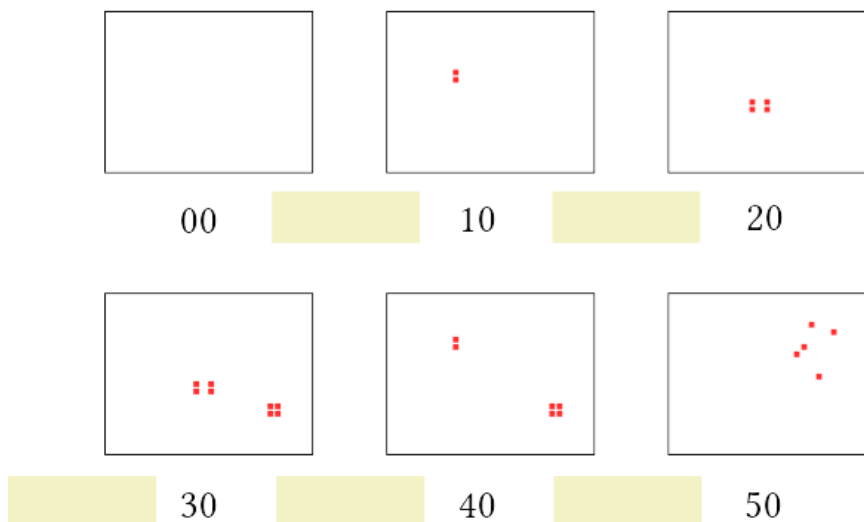
A: Tom berjumpa dengan Tina.

B: Seseorang membuka pintu gedung.

C: Tom dan Tina berjalan bergandengan.

D: Ada angin kencang.

E: Seseorang menutup pintu gedung.



Tantangan: Tentukan kejadian (A,B,C,D,E) pada kotak kuning . Tuliskan sebagai sebuah string terdiri dari huruf ABCDE sesuai dengan urutan kejadian. Contoh jawab: EDCBA

Jawaban:

Ada dua jawaban yang benar: **BACED** dan **EACBD**.

Antara detik 0 dan 10: Seseorang membuka pintu balai kota.

Pada gambar perbedaan pada detik ke-10, terdapat persegi merah pada posisi pintu yang menunjukkan perubahan, yang bisa disebabkan oleh membuka pintu (atau menutup pintu).

Antara detik 10 dan 20: Tom bertemu Tina.

Persegi merah di tengah gambar perbedaan pada detik ke-20 menunjukkan posisi di mana Tom dan Tina bertemu. Mereka mendekati satu sama lain dari arah kiri dan kanan sehingga mereka tidak terlihat di foto 10 detik sebelumnya.

Antara detik 20 dan 30: Tom dan Tina berjalan bersama dengan tangan saling berpegangan.

Gambar perbedaan pada detik ke-30 menunjukkan persegi merah yang menandakan posisi saat ini dan posisi sebelumnya dari Tom dan Tina. Saat berjalan bersama, mereka sangat dekat sehingga gambar kamera menggambarkan mereka sebagai satu titik.

Antara detik 30 dan 40: Seseorang menutup (atau membuka) pintu balai kota, sementara Tom dan Tina terus berjalan, meninggalkan pasar.

Pada gambar perbedaan pada detik ke-40, terdapat persegi merah pada posisi pintu yang menunjukkan perubahan, yang bisa disebabkan oleh menutup pintu (atau membuka pintu), dan pada posisi sebelumnya dari Tom dan Tina, yang tidak berada di sana lagi.

Antara detik 40 dan 50: Angin mulai bertiup.

Pada gambar perbedaan akhir pada detik ke-50, terdapat persegi merah di area tajuk pohon. Ini menunjukkan perubahan (seperti daun bergerak) yang mungkin disebabkan oleh angin.

Harap dicatat: Peristiwa B dan E dapat ditukar, karena tidak membuat perbedaan dalam gambar perbedaan.

Ini Informatika!

Pengolahan dan analisis gambar otomatis adalah fitur penting dalam sistem keamanan di tempat umum seperti bandara dan stasiun kereta api. Ini dapat digunakan untuk mendeteksi intruder di area terlarang atau mengidentifikasi tersangka dalam suatu penyelidikan. Namun, ini dapat menyebabkan hilangnya privasi jika tempat umum selalu dipantau.

Contoh tersebut mengilustrasikan bahwa memungkinkan untuk mendapatkan informasi dari gambar webcam menggunakan program komputer. Tugas analisis gambar bisa sesederhana membaca kode QR atau kode batang dari produk di supermarket, atau sekompleks memperkirakan jenis kelamin dan kelompok usia seseorang dari gambar wajahnya.

Sebagai gantinya, gunakan rekursi seperti berikut: Para leluhur Anda adalah: orang tua Anda dan leluhur mereka. Bagi Anda mungkin ini terdengar sebagai deskripsi yang aneh, tetapi bagi komputer hal ini sangat jelas.

Authorship

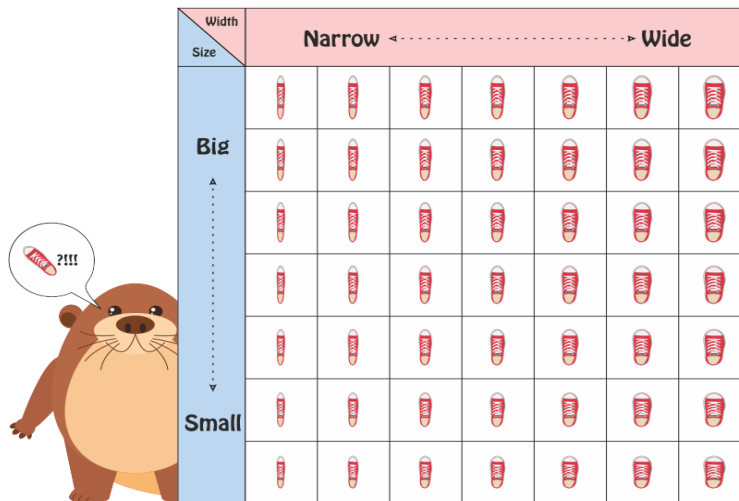
- Author: Michael Weigend (Germany), mw@creative-informatics.de, 2019-03-21; Contributors: Humberto Sermenó (El Salvador), humberto.sermenó@joventalento.edu.sv and Takeharu Ishizuka (Japan), isizukat@ishizukalab.net, 2019-05-14

License

Copyright © 2019 Bebras – International Contest on Informatics and Computer Fluency.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>

Bebras Koko pergi ke sebuah toko untuk membeli sepatu. Ia melihat beberapa pasang sepatu diatur sesuai ukuran dan lebarnya, mulai dari paling kecil dan sempit, berjajar lebih besar dan lebar, yang paling besar dan paling lebar ada di sudut paling kanan atas.



Karena Koko pelupa, ia lupa ukuran sepatunya, dan harus mencoba yang ukuran dan lebarnya paling pas. Koko menggunakan cara yang menjamin bahwa ia dapat menemukan sepatu paling pas dengan mencoba N kali.

Tantangan: Berapa nilai N yang paling kecil?

Pilihan Jawaban:

- a) 1
- b) 2
- c) 4
- d) 6

Jawaban:

Jawaban yang tepat adalah 2.

Bebras Koko bisa saja beruntung karena menemukan sepatu pada percobaan pertama. Namun, dapat dijamin bahwa ia dapat menemukan sepatu yang pas setelah mencoba 2 pasang sepatu.

Bebras Koko dapat memulai dari sepatu yang paling tengah seperti pada gambar berikut.

Width Size	Narrow <-----> Wide						
Big ↑							
Small ↓							

- Sepatu yang di tengah itu mungkin ukurannya sama, lebih kecil, lebih besar dan juga lebarnya pas, lebih sempit atau lebih lebar. Setelah mencoba, Koko akan berada pada salah satu situasi yang digambarkan dengan 9 zona yang diwarnai berbeda sebagai berikut.
- Jika sepatu itu pas untuknya, dia telah mendapatkan sepatu yang dicarinya
- Jika sepatu itu kekecilan dan terlalu lebar, ia akan mencari di zona 1
- Jika sepatu itu terlalu kecil tetapi sudah pas lebarnya, ia akan mencoba sepatu di zona 2
- Jika sepatu itu terlalu besar dan terlalu sempit, ia akan mencoba sepatu di zona 9 dan seterusnya

Width Size	Narrow <-----> Wide						
Big ↑	A	B					
		1		2		3	
		4		5		6	
Small ↓							
		7		8		9	

Misalnya, sepatu yang dicoba oleh Koko terlalu besar dan terlalu sempit, Koko akan mencoba sepatu yang lebih kecil dan lebih lebar, yaitu sepatu-sepatu di zona 1.

Koko akan mencoba sepatu yang paling tengah yaitu di zona 1 (yang sel yang ada tanda angka 1).

- Jika sepatu itu pas ukurannya, ia menemukan sepatu yang dicarinya
- Jika sepatu itu masih kekecilan dan terlalu lebar, sepatu di posisi A adalah sepatu yang cocok.
- Jika sepatunya kekecilan tapi pas lebarnya, sepatu di posisi B adalah sepatu yang cocok.

Seperti kita lihat, Koko harus mencoba paling banyak 2 pasang sepatu untuk menemukan sepatu yang cocok.

Jika Koko mulai dari posisi lainnya, ia harus mencoba lebih dari 2 kali.

Ini Informatika!

Pada soal di atas, sepatu yang dipajang di toko sepatu disusun dalam urutan peningkatan ukuran dan lebar, yang ditata pada dua dimensi. Untuk mendapatkan data dalam susunan teratur seperti itu disebut “sorting” (pengurutan). Algoritma pencarian biner digunakan pada data yang telah teratur untuk menemukan suatu nilai dengan sangat cepat. Setiap kali mencari, Pencarian biner mengurangi ruang pencarian menjadi setengahnya untuk mendapatkan jawaban yang benar, sehingga membutuhkan upaya yang lebih sedikit.

Menggunakan algoritma pencarian biner untuk menebak angka antara 1 dan 100 dapat mengarah pada menebak angka dalam upaya minimum.

Persoalan ini terkait dengan “binary searching” untuk menemukan sebuah nilai yang diinginkan, dalam suatu himpunan data yang disimpan dalam dua dimensi.

Authorship

- 2019-04-01 Jihye Kim (South Korea), anaki.csl@gmail.com
Task Proposal
- 2019-05-15: Eslam AbdElAal (Egypt) (eslam@aast.edu), Georgios Fesakis (Greece) (gfesakis@aegean.gr), Vipul Shah (India), v.shah@tcs.com

License

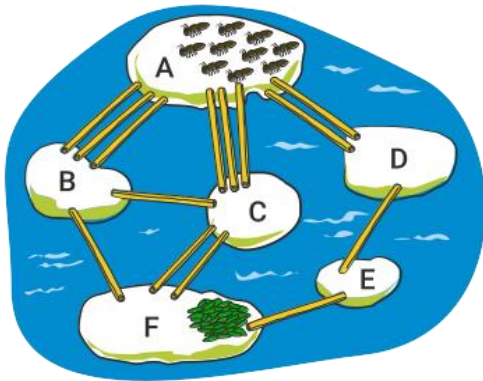
Copyright © 2019 Bebras – International Contest on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Semut di Sarang

PENEGAK (SMA)
2019-LT-06

10 semut sedang berada di lokasi A dan mencoba untuk menuju ke F. Hanya 1 semut dapat lewat batang kayu setiap saat. Untuk pergi dari satu lokasi menuju lokasi lainnya dibutuhkan 1 menit.



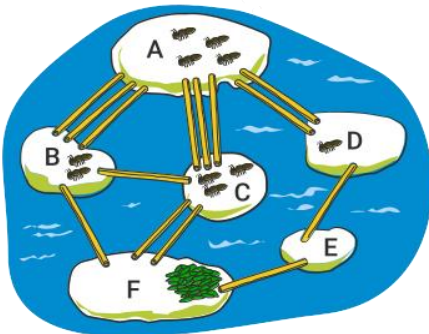
Tantangan: Berapa banyak semut dapat mencapai F dalam waktu 3 menit?
Jawab dengan bilangan bulat antara 0 s.d 10.

Jawaban:

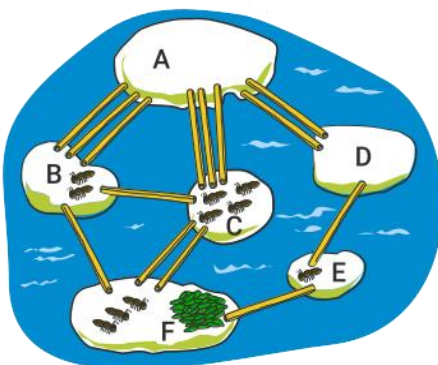
Jawaban yang tepat adalah **7**.

Situasi yang mungkin setelah setiap menit digambarkan sebagai berikut.

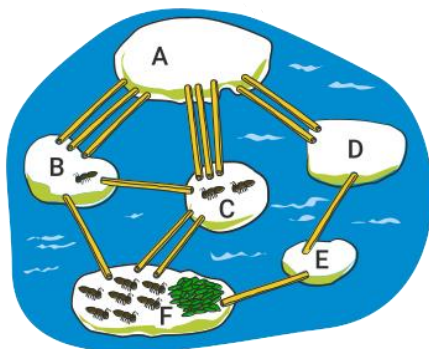
Setelah **satu** menit:



Setelah **dua** menit:



Setelah **tiga** menit:



Ini Informatika!

Tujuan yang ingin dicapai pada soal ini adalah untuk mengoptimalkan aliran semut melalui jaringan sehingga sebanyak mungkin semut sampai di lokasi makanan dalam waktu 3 menit. Masalah ini adalah masalah optimasi.

Semut yang tidak menyadari struktur jaringan yang mereka lalui tidak akan dapat menemukan solusi terbaik. Sebaliknya, seorang pengamat yang dapat melihat struktur jaringan dapat menemukan strategi yang optimal.

Pada soal ini, diasumsikan bahwa semut menyadari struktur jaringan, sehingga mereka dapat memilih jalur optimal untuk bergerak di sepanjang jalur tertentu.

Graf adalah struktur data abstrak yang digunakan untuk memodelkan jaringan dan terdapat banyak algoritma untuk mengoptimalkan aliran dalam keadaan tertentu. Atau tanpa menggunakan algoritma yang sudah ada sebelumnya, kami dapat menyatakan bahwa:

- Tidak masuk akal untuk mengirimkan lebih dari 1 semut lewat D-E.
- Tidak masuk akal untuk mengirimkan lebih dari 2 semut lewat A-B
- Saluran antara B-C tidak akan mempengaruhi aliran sehingga dapat diabaikan
- Faktor yang akan membatasi adalah B-F dan C-F

Pertimbangan tersebut memungkinkan kita untuk menemukan solusi optimal.

https://en.wikipedia.org/wiki/Flow_network

https://en.wikipedia.org/wiki/Maximum_flow_problem

https://en.wikipedia.org/wiki/Optimization_problem

Authorship

- Valentina Dagienė, Lithuania: valentina.dagiene@mii.vu.lt

License

Copyright © 2019 Bebras – International Contest on Informatics and Computational Thinking.

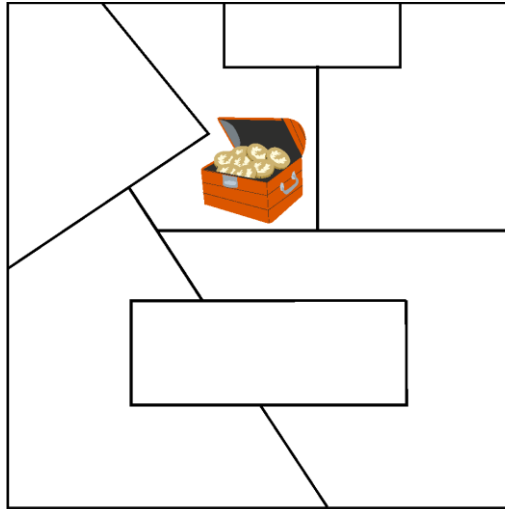
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Peta Yang Mana

PENEGAK (SMA)
2019-VN-04

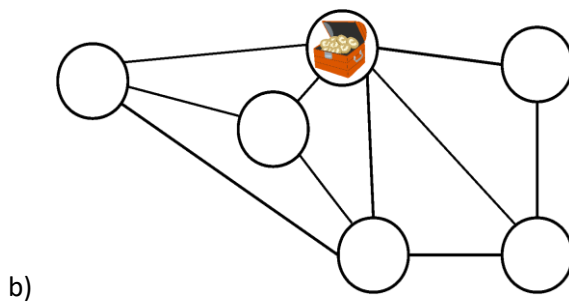
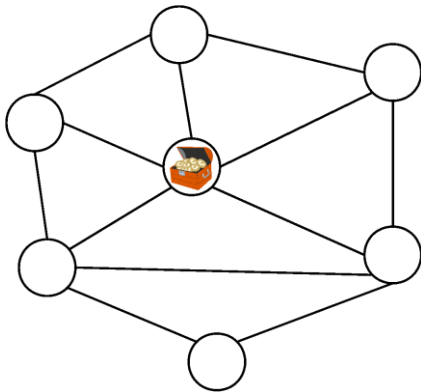
Sang Raja Bebras menyembunyikan harta karun di negara dengan 7 provinsi yang petanya ditunjukkan pada gambar sebagai berikut:

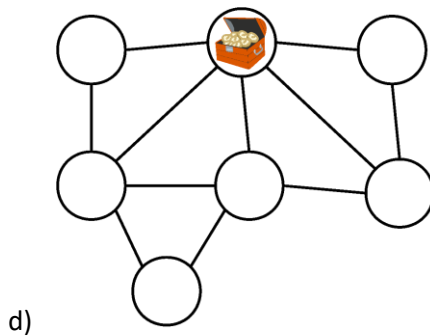
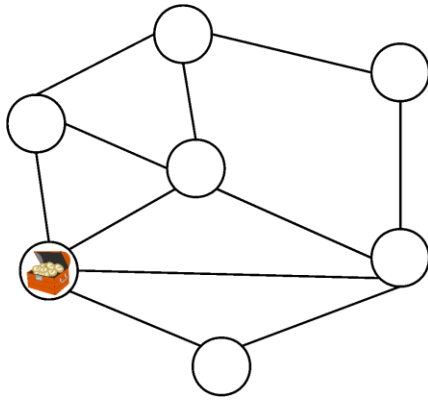


Supaya tidak mudah ditebak, Sang Raja menyimpan informasi peta tersebut dalam bentuk peta. Lingkaran merepresentasi provinsi dan garis penghubung menunjukkan dua provinsi yang bertetangga. Untuk mengecoh pencuri, Raja membuat 3 peta lain yang salah kodenya.

Tantangan: Peta mana yang merupakan peta yang benar?

Pilihan Jawaban:



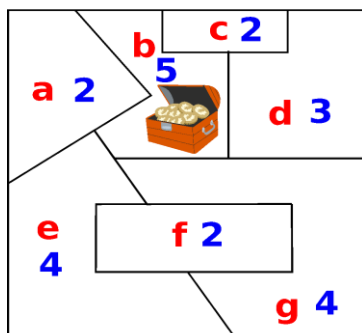


Jawaban yang tepat adalah: d)

Peta pada soal ini menunjukkan adanya 7 provinsi. Karena itu B tidak dapat dipilih sebagai jawaban yang benar karena graf B hanya memiliki 6 simpul.

Provinsi di mana harta karun terletak mempunyai 5 tetangga. Oleh karena itu C tidak dapat menjadi jawaban karena tidak ada simpul pada graf C yang mempunyai 5 jalur penghubung.

Untuk memutuskan apakah A atau D yang memenuhi syarat, kita perlu mencocokkan simpul dan jalur graf dengan peta. Kita terbantu dengan menghitung “tetangga” setiap simpul. Misalnya untuk provinsi a ke g (huruf dengan warna merah) dan menghitung tetangganya (warna biru):



Provinsi **a** memiliki 2 tetangga (**b** dan **e**).

Provinsi **b** memiliki 5 tetangga (**a**, **c**, **d**, **e**, dan **g**).

Provinsi **c** memiliki 2 tetangga (**b** dan **d**).

Provinsi **d** memiliki 3 tetangga (**b**, **c**, dan **g**).

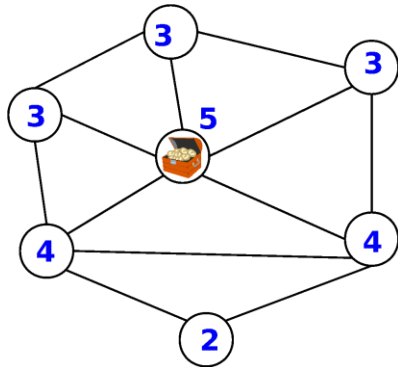
Provinsi **e** memiliki 4 tetangga (**a**, **b**, **f**, dan **g**).

Provinsi **f** memiliki 2 tetangga (**e** dan **g**).

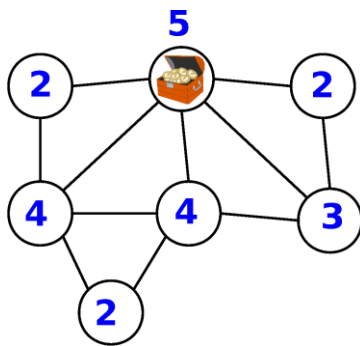
Provinsi **g** memiliki 4 tetangga (**b**, **d**, **e**, dan **f**).

Secara keseluruhan, kita memiliki tiga provinsi dengan 2 tetangga, satu provinsi dengan 3 tetangga, dua provinsi dengan 4 tetangga, dan satu provinsi dengan 5 tetangga.

Jika kita menghitung tetangga-tetangga di peta terenkripsi A, kita dapat melihat tiga provinsi dengan 3 tetangga dan hanya satu provinsi dengan 2 tetangga. Jadi A tidak bisa menjadi solusi yang benar.



Jika kita menghitung jumlah tetangga di peta terenkripsi D, kita dapat melihat bahwa jumlahnya sesuai dengan kebutuhan kita: tiga provinsi memiliki 2 tetangga, satu provinsi memiliki 3 tetangga, dua provinsi memiliki 4 tetangga, dan satu provinsi memiliki 5 tetangga.



Kita bisa berhenti di sini, karena kita telah menemukan solusi dengan mengeliminasi pilihan yang lain. Tetapi untuk memastikan, kita sebaiknya tetap mencocokkan lingkaran dengan negara-negara. Dan sebenarnya ini mungkin dilakukan seperti ini:

Ini Informatika!

Tugas ini menunjukkan contoh bagaimana graf dapat digunakan untuk merepresentasikan realitas. Sebuah graf memiliki simpul (lingkaran) dan tepian (garis) antara mereka. Tepian menunjukkan hubungan antara simpul-simpul tersebut. Dalam kasus ini, simpul-simpul tersebut menunjukkan provinsi-provinsi dan tepian menunjukkan hubungan "adalah tetangga dari". Graf merupakan alat yang sangat kuat ketika kita perlu mendeskripsikan dengan jelas hubungan antara beberapa entitas (provinsi, dalam kasus ini) dalam dunia nyata. Matematikawan dan ilmuwan komputer telah mengembangkan banyak algoritma yang berguna yang dapat diterapkan pada graf (contoh sederhana dalam tugas ini adalah menghitung jumlah simpul).

Untuk informasi lebih lanjut tentang graf, silakan lihat di sini:

[https://en.wikipedia.org/wiki/Graph_\(discrete_mathematics\)](https://en.wikipedia.org/wiki/Graph_(discrete_mathematics))

Authorship

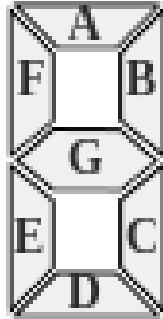
- Vu Van Luan and Le Anh Vinh, Vietnam, vuvanluanftu.k50@gmail.com, leanhvinh@gmail.com

- 2019-05-16 Peter Tomcsányi (Slovakia), tomcsanyi@slovanet.sk and Muhammad Faiz Ahmad Ismail (Malaysia), faiz@ardentedu.com: Reworked the original incomplete task idea.

License

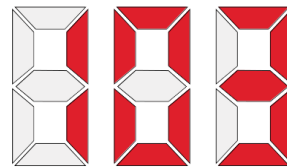
Copyright © 2019 Bebras – International Contest on Informatics and Computational Thinking. This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <https://creativecommons.org/licenses/by-sa/4.0/>

Alyna bermain angka dengan menggunakan LED (Light Emitting Diodes). Dia ingin menggunakan LED dengan 7 segmen untuk merepresentasikan sebuah angka. Setiap segmen LED ditandai dengan A, B, C, D, E, F, dan G, seperti ditunjukkan sebagai berikut.



Untuk memberitahukan segmen mana yang harus dinyalakan, ia harus menuliskan kode dalam sebuah tabel. Misalnya, tabel berikut akan menunjukkan 3 angka 103:

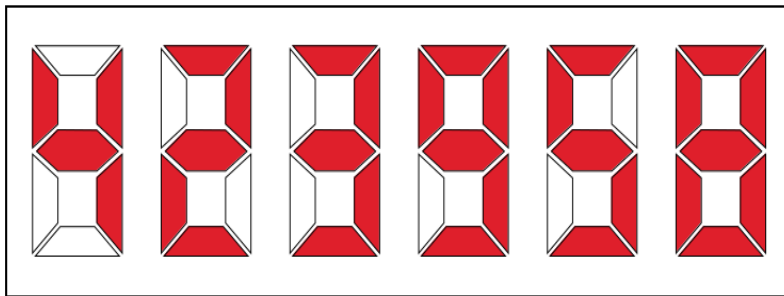
	A	B	C	D	E	F	G
↓							



Tantangan: Angka apa yang akan ditunjukkan oleh tabel sebagai berikut?

[illegible]

Jawaban yang tepat adalah:



Dari contoh tersebut, kita perhatikan bahwa setiap baris dalam tabel mewakili satu angka digital. Untuk baris pertama dalam tabel pertanyaan, hanya segmen **B, C, F, dan G** yang dipilih, yang mewakili angka 4. Untuk baris kedua, segmen **A, B, D, E, G** dipilih, mewakili angka 2. Melanjutkan proses ini untuk setiap baris, kita mendapatkan angka **423958**.

Ini Informatika!

Perubahan representasi satu objek dengan kumpulan objek lain disebut encoding. Encoding adalah hal yang sangat umum dalam Ilmu Komputer. Kita menggantikan setiap digit dengan urutan tujuh sel yang dapat memiliki dua nilai mungkin: merah atau putih.

Tampilan seperti ini banyak digunakan di era modern ini. Karena ukuran kecil dari dioda pemancar cahaya (LED), beberapa dioda dapat dihubungkan menjadi tampilan 7-segment.

Ketika kita menggunakan tampilan 7-segment, kita dapat mengirimkan urutan dari 7 nilai, baik angka satu (merah) atau angka nol (putih), untuk ABCDEFG di mana nol mematikan segmen dan satu menghidupkannya. Sebagai contoh, angka "4" dalam tampilan 7-segment dijelaskan oleh **0110011**.

Authorship

- 2019-04-01 Muhammad Faiz Bin Ahmad Ismail (MY), faiz@ardentedu.com
- Contributors:
 - 2019-05-15 Humberto Sermenó (El Salvador), humberto.sermenó@joventalento.edu.sv and Takeharu Ishizuka (Japan), isizukat@ishizukalab.net: text revisions, extended It's Informatics, replaced images with newest versions
 - 2019-05-15 Alisher Ikramov (Uzbekistan), ikramov.alisher@list.ru: It's Informatics section changes
 - 2019-05-16 WG4: Špela Cerar (SI) spela.cerar@pef.uni-lj.si, Mewati Ajub (ID), Takeharu Ishizuka (JP), Arnheidur Gudmundsdóttir (IS), Final polishing of the task (clarification of the task text, explanation, It's Informatics)
 - 2019-05-16 Humberto Sermenó (El Salvador), humberto.sermenó@joventalento.edu.sv: fixing typos

License

Copyright © 2019 Bebras – International Contest on Informatics and Computational Thinking.

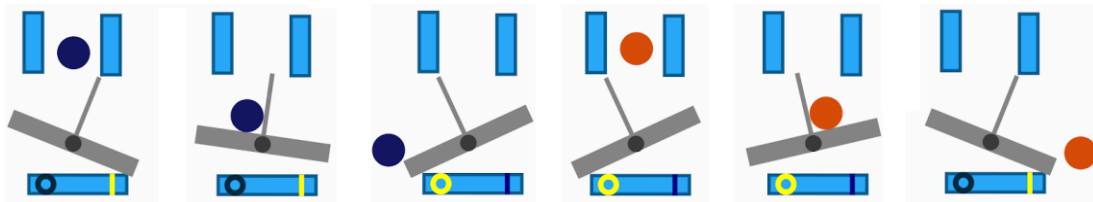
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>

	Pencacah	PENEGAK (SMA) 2019-AT-03
---	-------------------	-----------------------------

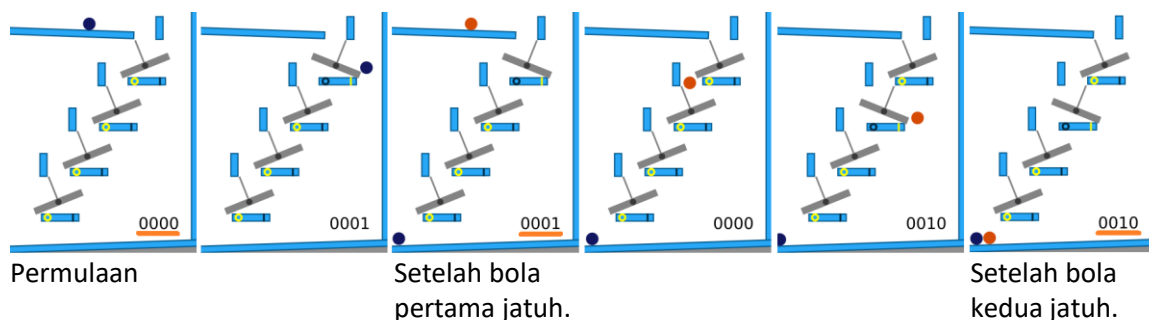
Pada mesin Pencacah ini, ada 4 batang yang dapat bergoyang miring ke kiri atau kekanan.

- batang bergoyang ke kiri= 0
- batang bergoyang ke kanan= 1

Saat sebuah bola jatuh pada sebuah batang, batang menjadi miring dan bola jatuh menggelinding. Ilustrasinya adalah sebagai berikut:



Berikut ini adalah mesin di mana dua bola dijatuhkan. Pada gambar pertama, semua batang miring ke kiri dan Pencacah menunjukkan 0000.



Tantangan:

Apa yang ditunjukkan pencacah setelah 5 bola dijatuhkan? Tuliskan jawaban dalam bentuk 4-digit yang terdiri dari 0 dan 1.

Jawaban:

Jawaban yang benar adalah 0101.

Urutan lengkapnya adalah: 0001, 0010, 0011, 0100, 0101.

Ini adalah penghitung biner mekanis.

- Posisi awalnya adalah 0000.
- Bola pertama membuat Batang 1 miring ke kanan. Penghitung menjadi 0001.
- Bola kedua membuat Batang 1 miring ke kiri dan Batang 2 miring ke kanan. Penghitung menjadi 0010.
- Bola ketiga membuat Batang 1 miring ke kanan dan kemudian berguling keluar. Batang 2 tetap miring ke kanan. Penghitung menjadi 0011.
- Bola keempat membuat Batang 1 miring ke kiri, Batang 2 miring ke kiri, dan Batang 3 miring ke kanan. Penghitung menjadi 0100.
- Bola terakhir membuat Batang 1 miring ke kanan dan kemudian berguling keluar. Batang 2 tetap miring ke kiri dan Batang 3 tetap miring ke kanan. Penghitung menjadi 0101.

Kita juga dapat melihatnya dari sudut pandang berikut:

Bola jatuh ke setiap batang sampai mencapai batang pertama yang berada dalam posisi 0. Lalu batang miring ke arah kanan dan keluar dari mesin.

Ini Informatika!

Ketika Anda terbatas untuk menghitung dengan hanya dua digit, yaitu 0 dan 1, Anda menggunakan sistem bilangan berbasis 2. Sistem ini disebut sistem biner. Biner digunakan oleh komputer karena setiap saklar di dalamnya hanya bisa diaktifkan (1) atau dimatikan (0). Komputer saat ini memiliki banyak saklar yang mengontrol semua informasi yang digunakan oleh komputer.

Mesin yang mudah dibangun ini mewakili penghitung biner mekanis. Penghitung ini memiliki 4 digit, sehingga dapat menghitung hingga angka 15.

Berikut adalah tabel dari semua angka 4 digit.

Binary number	Number
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

Berikut adalah video tentang bagaimana penghitung biner bekerja:

https://www.youtube.com/watch?v=zELAfmp3fXY&fbclid=IwAR1iVWNrmrj-bNzD26i5xNivAa2Ue7o_AWkRtlpRGamiHFEsOfigpRWizQ

Authors and Contributors (incl. Graphics)

- Wilfried Baumann (Austria), baumann@ocg.at
- 2019-05-15 - Jiří Vaníček (CZ), vanicek@pf.jcu.cz, Laura Braun (NO), laura.braun@fosenskole.no, Linda Björk Bergsveinsdóttir (IS), linda@sky.is
- Bebras Workshop

License

Copyright © 2018 Bebras – International Contest on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>