



Tim Olimpiade Komputer Indonesia

TANTANGAN
Bebras Indonesia 2018
BAHAN BELAJAR COMPUTATIONAL THINKING

Tingkat SMP



<http://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 kedua bulan November.

Tantangan Bebras Indonesia dibagi dalam 3 kategori, yaitu:

1. Siaga, untuk siswa SD dan yang sederajat
2. Penggalang, untuk siswa SMP dan yang sederajat
3. Penegak, untuk siswa SMA dan yang sederajat.

Pada Tantangan Bebras 2018, untuk kategori Siaga (SD) diberikan 12 soal yang harus diselesaikan dalam waktu 45 menit. Untuk kategori Penggalang (SMP) dan Penagak (SMA) masing-masing diberikan 15 soal yang harus diselesaikan dalam waktu 45 menit.

Tantangan Bebras Indonesia 2018 dapat berjalan lancar berkat dukungan penuh dari 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 <http://bebras.or.id>

Selain dari itu para Koordinator Bebras Biro dan tim yang tersebar di lebih dari 40 perguruan tinggi di seluruh Indonesia yang langsung berhubungan dengan para siswa dalam menyelenggarakan Tantangan Bebras Indonesia 2018.

Penyiapan soal-soal dan pengelolaan Tantangan Bebras Indonesia 2018 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), Cecilia Nugraheni dan Vania Natalia (Unpar), serta penyiapan buku ini dibantu oleh Inez Perera, dan Rana R. Natawigena.

Bahan belajar Computational Thinking Tantangan Bebras Indonesia 2018 ini dibagi dalam tiga buku sesuai kategori, yaitu buku untuk Tingkat SD (Siaga), Tingkat SMP (Penggalang), dan Tingkat SMA (Penegak).

Karya ini dilisensikan di bawah lisensi Creative Commons Attribution-NonCommercial-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.
- NonCommercial: 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.

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. Di bidang "*Computing*", yang diterjemahkan ke dalam bahasa Indonesia sebagai 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 yaitu bisa berupa "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 menyaring bagian penting dari suatu permasalahan dan mengabaikan yang tidak penting, sehingga memudahkan fokus kepada solusi
- Algoritma, yaitu menuliskan otomasi solusi melalui berpikir algoritmik (langkah-langkah yang terurut);
- Pengenalan pola persoalan, generalisasi serta mentransfer proses penyelesaian persoalan ke sekumpulan persoalan sejenis.

Secara operasional, keempat fondasi berpikir tersebut dijabarkan lagi menjadi definisi operasional yang didefinisikan oleh CSTA¹ yaitu:

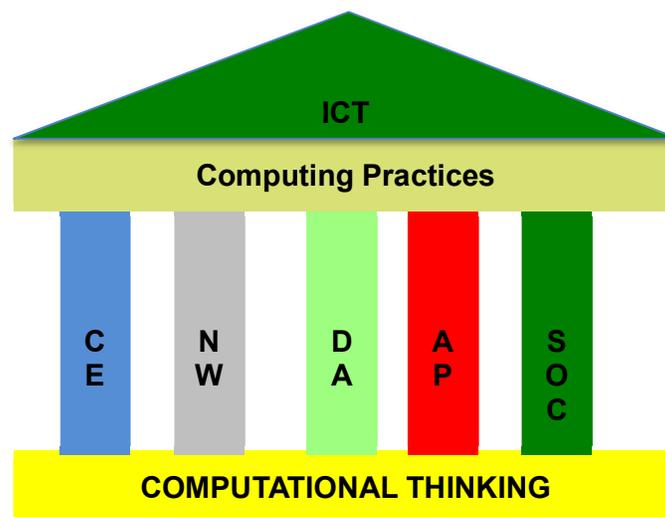
- Memformulasikan persoalan sehingga dapat menentukan solusinya, baik yang akan diselesaikan dengan bantuan komputer, atau *tools* lainnya
- Meng-organisasikan dan menganalisis data secara logis;
- Merepresentasikan data melalui abstraksi dalam bentuk model, dan melakukan simulasi;
- Melakukan otomasi solusi dengan menyusun algoritma;
- Mengidentifikasi, menganalisis, dan mengimplementasi solusi yang mungkin diperoleh, dengan tujuan agar langkah dan sumberdayanya efisien dan efektif.
- Melakukan generalisasi dan mentransfer proses penyelesaian persoalan untuk dapat menyelesaikan persoalan-persoalan yang sejenis.

Kemampuan dan ketrampilan berpikir komputasional tersebut ditunjang dengan beberapa sikap sebagai berikut:

- Yakin dan percaya diri dalam menghadapi dan mengelola kompleksitas.
- Gigih dan tekun bekerja dalam menghadapi persoalan yang sulit.
- Toleran terhadap ambiguitas.
- Kemampuan untuk menangani "open ended problems".
- Kemampuan berkomunikasi dan bekerjasama dalam tim untuk mencapai suatu tujuan atau menghasilkan solusi.

¹ <https://id.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf>

Di negara maju, “*Computer Science*” (yang di Indonesia juga 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 (CE), Jaringan Komputer (NW), Analisis Data (DA), Algoritma dan Pemrograman (AP), dan Aspek Sosial dari pemanfaatan Informatika (SOC). Selain pengetahuan, juga didefinisikan praktek-praktek komputasi untuk mengemas pengetahuan dan memraktekannya, yaitu: pembinaan menumbuhkan budaya komputasi, menciptakan artifak, berkolaborasi untuk mewujudkan suatu produk TIK, menguji dan memperbaiki/menyempurnakan artefak TIK, mengenali dan mendefinisikan problema-problema komputasi, berkomunikasi tentang komputasi, dan mengembangkan serta menggunakan abstraksi.



Gambar 1. Hubungan Computational Thinking, Informatika dan TIK

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.

Computational Thinking juga dapat dilatih melalui kegiatan-kegiatan yang mengekspresikan cara berpikir, misalnya koding dan pemodelan sistem yang kemudian disimulasi.

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 ketrampilan

menggunakan gadget dan aplikasinya, tetapi juga kemampuan untuk menggunakan dan memanfaatkan konten dengan bijak.

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.

Computational Thinking sudah menjadi salah satu kemampuan yang termasuk diujikan sebagai bagian dari test matematika PISA mulai tahun 2021. Oleh karena itu, marilah berlatih computational thinking dari sekarang, salah satunya dengan mengikuti Tantangan Bebras.

Tantangan Bebras

(Bebras Computational Thinking Challenge)

Situs: <http://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 dan sudah diikuti oleh sekitar 2,75 juta siswa yang berasal dari 60 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” ketrampilan berpikir untuk menyelesaikan persoalan, yaitu melalui pendekatan *constructionism* yang diperkenalkan oleh Seimort 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 Untuk mulai tertarik ke topik-topik informatika dan memecahkan persoalan dengan menggunakan informatika
- Men-stimulasi minat siswa ke informatika
- Mendorong siswa untuk menggunakan “TIK” dengan lebih intensif dan kreatif dalam aktivitas belajarnya
- Menyemangati siswa untuk berpikir lebih dalam dari pada sekedar ke komputer/alatnya dan TIK.

Tantangan bebras diselenggarakan sekali setahun pada saat hampir bersamaan di seluruh dunia, sepanjang pekan Bebras, yang ditetapkan pada minggu kedua bulan November.

Bebras Indonesia

Situs: <http://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. Tantangan Bebras di Indonesia dilaksanakan secara online. Penyelenggaraan tantangan 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 supporter. 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.

Latihan Online

Latihan online disediakan di situs yang dapat diperoleh informasinya di situs <http://bebras.or.id>

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 SMP Tantangan Bebras 2018



DAFTAR SOAL

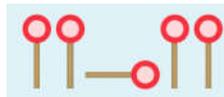
- Kode Pesan dari atas bukit (halaman 10)
- Menghubungkan Lingkaran (halaman 12)
- Mobil yang hilang (halaman 14)
- Soundex (halaman 16)
- Yahtzee (halaman 18)
- Jalan-jalan di Taman (halaman 21)
- Mutasi Mahluk Luar Angkasa (halaman 23)
- Membangun Bendungan (halaman 25)
- Kereta Api di Kota Bebras (halaman 27)
- Kode Bendera (halaman 29)
- Tiga sekawan Berang-berang (halaman 31)
- Bendungan (halaman 33)
- Tugas Satu Jam (halaman 35)
- Jembatan Antar Pulau (halaman 37)
- Kue Ulang Tahun (halaman 39)
- Berbagi Kamar (halaman 40)
- Buaya Bersisik (halaman 42)
- Petak dan Kotak (halaman 44)
- Kotak Tiga Berang-berang (halaman 46)
- Antrian Mobil (halaman 48)
- Infinite Ice Cream (halaman 50)
- Balon Ulang Tahun (halaman 52)
- Tempat Perlindungan (halaman 54)
- Pin Merah Putih (halaman 55)
- Kamping (halaman 56)
- Kelas Olah Raga (halaman 58)
- Danau Berang-berang (halaman 60)
- Twist and Turn (halaman 62)
- Mengunjungi Rumah Teman (halaman 64)
- Kotak Krayon (halaman 66)



Dua berang-berang, si Jana dan Robi bermain melempar pesawat terbang kertas. Salah satu dari mereka berada di atas bukit, dan yang lainnya berada di kaki bukit untuk memungut pesawat yang sudah mendarat. Malangnya, hanya berang-berang di atas bukit yang dapat melihat posisi pesawat mendarat. Mereka menciptakan kode dengan sebuah raket, yang memberikan kode kepada berang-berang di bawah bukit, kemana harus mengambil pesawat yang tiba

Kiri	Kanan	Mendekat ke bukit	Menjauh dari bukit

Ternyata, ada masalah dengan kode ini. Jika sebuah perintah dikirim tanpa jeda dengan perintah sebelumnya, bisa menimbulkan arti ganda. Misalnya, jika dikirimkan pesan sbb.



artinya: **Kiri, Mendekat ke bukit, Kiri**. Tapi juga dapat berarti: **Kiri, Kanan, Kiri, Kiri**. Jana dan Robi harus mengusulkan satu tabel kode yang tidak menimbulkan arti ganda **untuk pesan di atas**, dari 4 tabel kode berikut:

Tantangan:

Mana tabel kode yang tidak menimbulkan arti ganda untuk pesan di atas?

Pilihan Jawaban:

- A.

Kiri	Kanan	Mendekat ke bukit	Menjauh dari bukit
- B.

Kiri	Kanan	Mendekat ke bukit	Menjauh dari bukit
- C.

Kiri	Kanan	Mendekat ke bukit	Menjauh dari bukit
- D.

Kiri	Kanan	Mendekat ke bukit	Menjauh dari bukit

Jawaban:

Kiri	Kanan	Mendekat ke bukit	Menjauh dari bukit

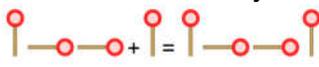
Jawaban yang benar adalah **C**

Pertama-tama kita sebaiknya melihat apakah ada jawaban lain yang benar.

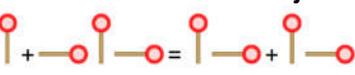
Jawaban B bukan kode yang baik karena **Kiri** diikuti oleh **Kanan** menggunakan kode yang sama dengan

Mendekat ke bukit + =

Jawaban D bukan kode yang baik karena **Kiri** diikuti oleh **Menjauh dari bukit** yang menggunakan kode

yang sama dengan **Mendekat ke bukit**. 

Sedikit lebih sulit untuk melihat apakah **jawaban A juga tidak benar**. Di sini **Kiri** diikuti oleh **Menjauh dari**

bukit yang menggunakan kode yang sama dengan **dua kali Kanan**. 

Jadi bagaimana kita dapat yakin bahwa C adalah kode yang baik dan bahwa kita tidak akan pernah menemukan dua urutan perintah berbeda yang menghasilkan serangkaian sinyal yang sama? Perhatikanlah bahwa keempat perintah dikodekan sebagai kode yang naik diikuti oleh kode nol atau kode yang lebih rendah. Jadi, begitu Robin melihat kode yang muncul, dia tahu bahwa kode kata baru telah dimulai. Untuk memecahkan kode suatu pesan, ia hanya perlu menghitung jumlah kode yang diturunkan diantara kode yang muncul berikutnya

Ini Informatika!

Ketika komputer saling mengirim pesan atau perintah melalui kabel, atau melalui udara di jaringan nirkabel (*wireless*), pengiriman pesan ini dilakukan dengan menggunakan urutan sinyal yang cepat, di mana untuk setiap sinyal ada dua kemungkinan (berpikir menyala atau mati tetapi seringkali sedikit lebih rumit), sama seperti Jana dan Robin menggunakan dua sinyal yang berbeda untuk pesan mereka.

Cara spesifik dimana pesan atau perintah diterjemahkan ke dalam sinyal, disebut kode (biner). Dalam hal ini, panjang kode biasanya berbeda-beda, karena jumlah sinyal yang digunakan untuk satu pesan atau perintah tidak harus selalu sama.

Yang penting adalah sang penerima pesan dapat menerjemahkan sinyal kembali ke pesan asli tanpa membuat kesalahan. Dengan kata lain, anda harus berhati-hati ketika merancang kode seperti itu. Kode yang 'baik' adalah kode unik yang dapat di-de-kodekan.

Jenis khusus kode yang dapat di-de-kodekan secara unik adalah kode awalan. Ini adalah kode di mana tidak ada kata lain yang dimulai dengan urutan penuh sinyal yang sama atau yang mempunyai simbol yang sama dengan kode yang lain, seperti dalam contoh berikut:

Kode awalan memiliki sifat yang bagus: mereka dapat dibuat cukup pendek dan mudah di-de-kodekan. Mereka tidak hanya digunakan untuk keperluan komunikasi tetapi juga muncul dalam beberapa algoritma kompresi (dan dalam berbagai tugas Bebras).

Authorship

2018-04-02 Nils Mak (BE): original idea - used prefix codes instead of uniquely decodable codes

2018-04-06 Kris Coolsaet (BE), kris.coolsaet@ugent.be: Major edit + Task Proposal

2018-05-03 Kris Coolsaet (BE), kris.coolsaet@ugent.be: Small revision in reaction to the reviews

2018-05-09 Complete revision by Workgroup 11, baumann@ocg.at, Austria

License

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

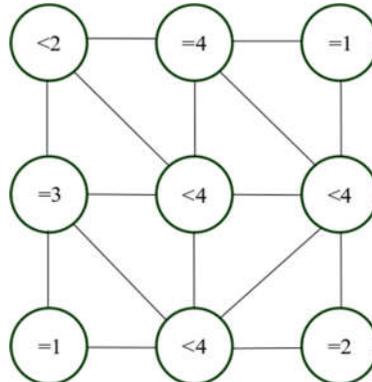
This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).



Menghubungkan Lingkaran

PENGGALANG (SMP)
I-2018-CA-05

Tugas anda adalah mewarnai lingkaran-lingkaran pada gambar berikut. Lingkaran-lingkaran tersebut dihubungkan dengan lingkaran tetangganya (yang terhubung langsung dengan garis). Terdapat 9 lingkaran dan 16 hubungan antar dua buah lingkaran. Angka yang dituliskan dalam lingkaran menunjukkan jumlah tetangga yang harus diwarnai. Misalnya sebuah lingkaran dengan tulisan “=3”, artinya 3 dari 4 tetangganya harus diwarnai. Sebuah lingkaran dengan tulisan “<4” artinya lingkaran tetangga yang harus diwarnai kurang dari 4.



Tantangan:

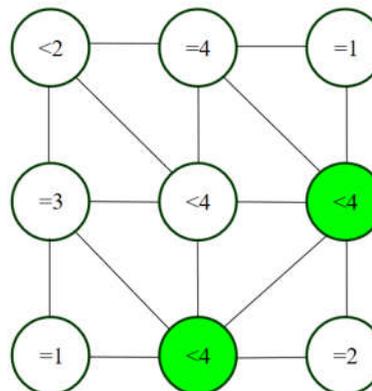
Berapa banyak lingkaran yang harus anda warnai?

Pilihan Jawaban:

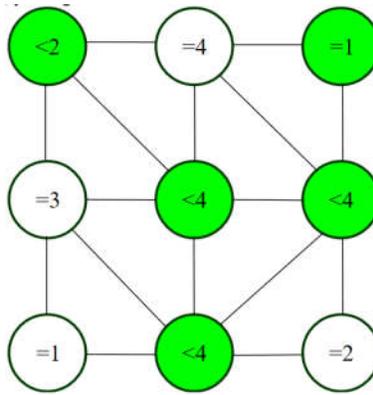
- A. 4
- B. 5
- C. 6
- D. 7

Jawaban:

Dengan mencermati lingkaran paling kanan bawah yang berisi “=2”, kita tahu bahwa tetangganya harus diisi untuk memperoleh:



Dengan mencermati lingkaran yang isinya “=4”, 4 tetangganya harus diisi sehingga



Dengan demikian, semua lingkaran sudah memenuhi kriteria. Mencermati setiap lingkaran sisanya, kita lihat bahwa lingkaran-lingkaran tersebut tidak dapat diwarnai. Secara lebih spesifik:

Jika lingkaran "=1" diwarnai, maka lingkaran "=3" menjadi salah

Jika lingkaran "=2" diwarnai, maka lingkaran "<4" di atasnya menjadi salah

Jika lingkaran "=3" diwarnai, maka lingkaran "<2" di atasnya menjadi salah

Jika lingkaran "=4" diwarnai, maka lingkaran "=1" di kanannya menjadi salah

Catatan: kita juga bisa mulai dari lingkaran "=4" dan akan mendapatkan hasil yang sama dengan penalaran yang sama.

Inilah Informatika!

Masalah ini memerlukan logika dan menunjukkan bahwa solusi brute-force tidak efektif. Kalau kita berusaha mencari jalan keluar dengan semata-mata menggunakan kekuatan pemrosesan komputer, perhatikan bahwa untuk masing-masing dari 9 lingkaran, ada yang diwarnai dan ada yang tidak. Jadi ada 2 kemungkinan untuk setiap lingkaran. Jumlah total kemungkinan untuk mewarnai lingkaran ini adalah $2^9 = 512$. Jadi, salah satu solusinya adalah mencoba semua 512 cara mewarnai lingkaran ini. Namun, dengan menggunakan penalaran, khususnya, urutan pengurangan yang sesuai logika seperti yang ditunjukkan dalam solusi, secara substansial mengurangi jumlah kemungkinan.

Authorship

Troy Vasiga, troy.vasiga@uwaterloo.ch, Canada Susanne Datzko, susanne@datzko.ch, Switzerland

Lisensi

Copyright © 2018 Bebras – International Contest on Informatics and Computer Fluency.

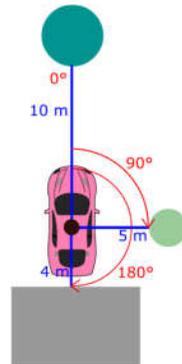
This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License. <http://creativecommons.org/licenses/by-nc-sa/3.0/>.



Mobil yang hilang

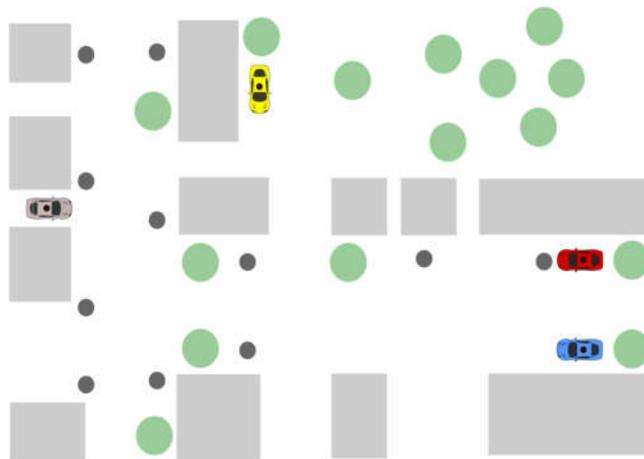
PENGGALANG (SMP)
I-2018-DE-03

Celaka! Sebuah mobil tanpa pengemudi tidak dapat kembali ke garasinya karena terhenti di sebuah lokasi di kota sebab baterai-nya habis. Sebelum baterai habis, berkat sensornya, mobil tersebut menemukan sebuah tempat parkir dan mengirimkan posisi objek yang ada di sekitarnya. Setiap objek mempunyai 2 nilai, yaitu sudut dan jarak dari mobil. Sudut diukur 360° menggunakan sensor di atas atap mobil. Sudut 0° adalah sisi depan mobil - lihat gambar.



Pada contoh, hasil sensor adalah $[(0,10),(90,5),(180,4)]$.

Sebuah mobil yang hilang mengirimkan hasil sensornya sebagai berikut: $[(0, 5), (90, 4), (180, 5), (270, 12)]$



Tantangan:

Mobil manakah yang hilang berdasarkan hasil sensor ini?

Pilihan Jawaban:

- A.
- B.
- C.
- D.

Jawaban:

Jawaban yang tepat adalah B. 

Mobil merah (di tengah sisi kanan pada peta) adalah mobil yang hilang.

- Mobil yang hilang memiliki benda 5m di depannya, benda 4m di sisi kanannya, benda 5m di belakangnya, dan benda dengan jarak 12m di sisi kirinya.
- Mobil abu-abu bukanlah mobil yang hilang, karena benda-benda di sisi kanan dan kirinya mempunyai jarak yang sama dekatnya.
- Mobil kuning bukanlah mobil yang hilang, karena objek di belakangnya (lingkaran hijau) jauh lebih dekat daripada objek di depannya (blok abu-abu).
- Mobil biru bukanlah mobil yang hilang, karena objek di depan (lingkaran hijau) jauh lebih dekat daripada objek di belakang (lingkaran hitam kecil).

Mobil merah adalah mobil yang hilang: objek di depan dan di belakang sama-sama dekat, dan objek di sisi kanan (blok abu-abu) jauh lebih dekat daripada objek di sisi kiri (mobil biru).

Ini Informatika!

Mobil tanpa pengemudi menggunakan metode LIDAR (deteksi cahaya dan jangkauan) untuk menandai lingkungan mereka, dengan memakai teknologi laser. Perangkat lunak navigasi mereka menciptakan model 3D yang kompleks dari semua objek hingga jarak beberapa ratus meter.

Sebaliknya, model dalam tugas ini sangat sederhana, dan hanya merekam objek terdekat.

Secara umum, model adalah abstraksi realitas, dan hanya mencakup aspek-aspek realitas yang penting untuk tujuan tertentu. Dalam hal mengemudi secara otomatis, model ini dapat digunakan untuk pencegahan tabrakan. Untuk tujuan ini, lokasi relatif benda-benda di sekitarnya diperlukan. Aspek lain seperti warna tidak penting dan dapat diabaikan.

<https://en.wikipedia.org/wiki/Lidar>.

Authorship

2018-04-06 Michael Weigend (Germany), mw@creative-informatics.de:

License

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

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

Bob si berang-berang mengembangkan sebuah sistem pengkodean yang terdiri dari 4 digit, untuk mengkode sebuah kata menjadi kata rahasia:

1. Huruf pertama selalu diambil.
2. Hapus semua huruf 'A', 'E', 'I', 'O', 'U', 'H', 'W', 'Y'.
3. Ganti huruf menjadi sebuah angka dengan aturan:

Huruf	Menjadi Angka
B, F, P, V	1
C, G, J, K, Q, S, X, Z	2
D, T	3
L	4
M, N	5
R	6

4. Ganti dua atau lebih huruf yang muncul berturut-turut dengan huruf tersebut diikuti jumlah kemunculannya.
5. Ambil 4 digit pertama saja, tambahkan 0 jika kurang dari 4 untuk menjadi 4 digit.

Tantangan:

Kode apa yang diperoleh untuk kata **HILBERT**?

Pilihan Jawaban:

- A. H041
- B. B540
- C. H410
- D. H416

Jawaban yang benar: D. H416

- 1) Simpan huruf pertama dari kata (H).
- 2) Hilbert → H4163 (setelah menerapkan tiga aturan pertama).
- 3) H416 (hanya mengambil empat posisi pertama).

Ini Informatika!

Algoritma ini dikenal sebagai Soundex.

Soundex adalah algoritma fonetik untuk mengindeks nama dengan suara, sebagaimana diucapkan dalam bahasa Inggris. Algoritma semacam ini digunakan dalam pencarian. Teknik ini berkaitan dengan koreksi fonetik: kesalahan ejaan yang muncul karena pengguna mengetik kueri (serangkaian kata) yang terdengar seperti kata target. Algoritma ini bisa digunakan untuk tugas "penarikan tinggi" (misalnya Interpol), tapi biasanya tidak bisa tepat untuk nama-nama negara tertentu.

<https://en.wikipedia.org/wiki/Soundex>

Algoritma ini dijelaskan oleh Donald Knuth dalam "The Art Of Computer Programming, vol. 3: Sorting And Searching".

Authorship:

2018-04-09 Ionuț Gorgos (Pakistan), ionut.gorgos@stud.acs.upb.ro

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/>

Yahtzee adalah permainan menggunakan 5 dadu, dan 3 kocokan dadu per putaran. Pada satu kocokan, pemain dapat memilih hanya sebagian dadu yang dikocok (tidak harus semuanya). Jika akhirnya sisi atas semua dadu bernilai sama, pemain memperoleh sebuah “Yahtzee” (menang).

Gambar berikut menunjukkan dadu yang tidak dikocok (“keep”) dan yang dikocok (warna hitam) dalam sebuah putaran:

Throw 1					Throw 2			Throw 3		Throws for next player				
1	6	5	6	5	6	6	5	6	6	1	5	5	1	1
	Keep		Keep		Keep			Keep	Keep					

Pada contoh tersebut, pemain mengocok dadu dengan permukaan {1,5,6} pada giliran pertama. Pemain tidak mengocok dadu dengan permukaan 4 sebab paling banyak.

Pada kocokan berikutnya, pemain mengocok tiga dadu sisanya dan tidak mengocok yang nilainya 1. Pada kocokan ketiga (yang terakhir), pemain hanya mengocok dua dadu dan berhasil mendapatkan 5 dadu dengan permukaan sama yang disebut Yahtzee.

Marti si berang-berang mencatat keadaan dadu yang dalam 3 kocokan berhasil mendapat Yahtzee. Malangnya, daftarnya tercampur dengan daftar berang-berang Muti yang juga mencatat.

Tantangan:

Dengan asumsi bahwa dia selalu menyimpan dadu yang paling banyak muncul pada setiap kocokan, daftar mana yang menunjukkan daftar Yahtzee?

Pilhan Jawaban:

- A.
- B.
- C.
- D.

Jawaban yang benar:

B.

Throw 1					Throw 2			Throw 3	Throws for next player					
1	1	6	6	5	6	5	6	6	1	1	5	1	5	1
		Keep	Keep		Keep		Keep	Keep						

Setelah kocokan pertama, pemain mendapat dua angka enam dan menyimpannya. Pada kocokan kedua, pemain melempar tiga dadu dan mendapat dua angka enam lagi. Pada kocokan terakhir, pemain hanya

melempar satu dadu dan menerima "6" yang terakhir dengan demikian mendapatkan "Yahtzee" yang terdiri dari lima enam.

Jawaban yang lain salah karena:

A.

Throw 1					Throw 2			Throw 3		Throws for next player				
		Keep	Keep				Keep	Keep						

Setelah kocokan pertama, pemain mendapat dua angka enam dan menyimpannya. Pada kocokan kedua, pemain melempar tiga dadu dan mendapat satu lagi angka enam. Pada i kocokan terakhir, pemain melempar dua dadu dan hanya menerima satu "6" sehingga dia tidak mendapatkan "Yahtzee".

C.

Throw 1					Throw 2			Throw 3		Throws for next player				
		Keep	Keep		Keep	Keep								

Setelah kocokan pertama, pemain mendapat dua angka enam dan menyimpannya. Pada kocokan kedua, pemain melempar tiga dadu dan mendapat angka enam dua kali lagi. Pada kocokan terakhir, pemain melempar satu dadu dan tidak mendapat "6" sehingga dia tidak mendapatkan "Yahtzee".

D.

Throw 1					Throw 2			Throw 3		Throws for next player				
		Keep	Keep				Keep		Keep					

Setelah kocokan pertama, pemain mendapat dua angka enam dan menyimpannya. Pada kocokan kedua, pemain melempar tiga dadu dan mendapat angka enam lagi. Pada kocokan terakhir, pemain melempar dua dadu dan hanya mendapatkan satu angka enam, sehingga dia tidak mendapatkan "Yahtzee".

Ini Informatika!

Ada banyak masalah seperti ini yang dapat dijelaskan oleh model input-processing-output (IPO). Input adalah urutan hasil dadu dari tiap kocokan yang harus diproses untuk menghasilkan sebagai output apakah "Yahtzee" dimenangkan atau tidak.

Contoh lain adalah evaluasi ekspresi, mis., "50 * 4". Untuk mengevaluasi ekspresi ini, komputer pertama kali membaca "50", dan menyadari bahwa ini pasti angka. Selanjutnya, ketika terbaca "*", ia mengerti bahwa angka tersebut harus dikalikan dengan sesuatu. Ketika membaca final "4", ia mendapatkan angka lainnya dan sekarang dapat melakukan perhitungan dan menampilkan hasilnya.

Authorship

2018-04-09 Lars-Åke Nordén (Sweden), lln@it.uu.se:

2018-05-08 Eljakim Schrijvers (USA), eljakim@bebraschallenge.org: rewrote the tasks, added the table for the example.

2018-05-08 Budit Thanasopon (Thailand), bundit@it.kmitl.ac.th: rewrote the tasks, added the table for the example

License

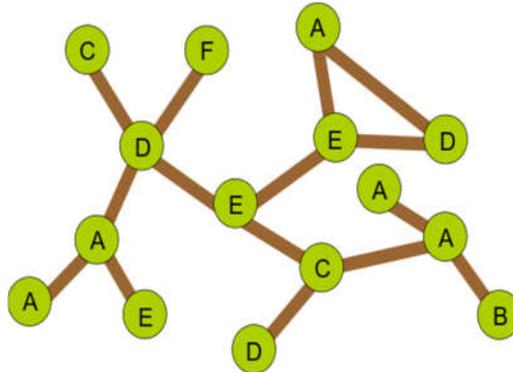
Do note that "Yathzee" is a trademark of Hasbro.

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

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



Berikut ini adalah peta untuk berjalan-jalan di taman Bebras:



Lingkaran hijau melambangkan pohon dan garis coklat melambangkan jalan. Perhatikan bahwa huruf yang sama dapat dipakai melambangkan pohon. Trayek dari pohon F ke pohon B dapat dituliskan sebagai F D E C A B. Pada suatu hari minggu, ada dua keluarga jalan-jalan di taman bebras.

Trayek keluarga Amir adalah B A A C E D E E D A.

Trayek keluarga Badu adalah F D C D A E A D E D A.

Tantangan:

Asumsikan bahwa kedua keluarga tersebut berangkat bersamaan dan berjalan dari satu pohon ke pohon berikutnya. Waktu tempuh dari satu pohon ke pohon berikutnya selalu sama. Berapa kali kedua keluarga tersebut bertemu pada pohon yang sama?

Pilihan jawaban:

- A. Dua kali
- B. Satu kali
- C. Tiga kali
- D. Mereka tidak pernah bertemu

Jawaban:

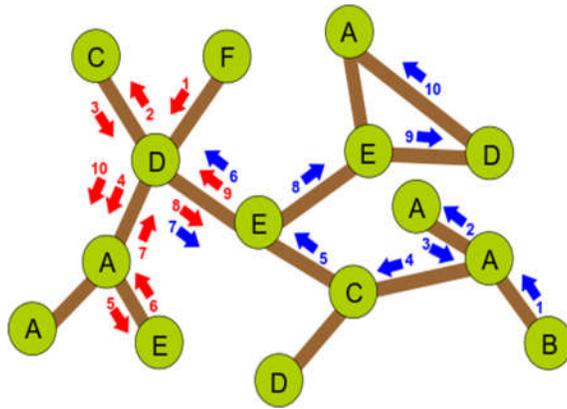
Jawaban yang benar adalah D. Mereka tidak pernah bertemu.

Kita tidak bisa hanya menemukan huruf yang sama di posisi yang sama (misalnya Waktu yang sama) selama perjalanan, karena huruf yang sama dapat menunjukkan pohon yang berbeda dari jenis yang sama. Misalnya kedua keluarga mengakhiri perjalanan mereka di pohon berlabel A, tetapi jika kita mengikuti langkah demi langkah mereka, kita akan mengetahui bahwa sebenarnya mereka berakhir pada pohon yang berbeda.

Perhatikan bahwa kita dapat mengikuti setiap langkah dari setiap keluarga dengan cukup mudah karena tetangga dari setiap pohon (pohon yang terhubung dengannya melalui jalur) selalu diberi label dengan huruf yang berbeda.

Tetapi untuk mengikuti dua jalan secara paralel tidak mudah (walaupun bisa dilakukan).

Jadi mari kita lukis awalnya jalan-jalan keluarga Amir (mulai dari B, dengan warna biru) dan beri nomor pohon yang dikunjungi pada saat mereka mengunjunginya (angka biru 1 hingga 11). Lalu mari kita melukis dengan cara yang sama dengan keluarga Badu yang berjalan dengan warna merah dan angka merah.



Kedua keluarga akan bertemu di beberapa pohon yang diberi nomor dengan warna merah dan biru. Tetapi kita tidak bisa melihat pohon seperti itu dalam gambar.

Perhatikan bahwa hanya ada dua pohon yang dikunjungi oleh kedua keluarga ini, jadi kita hanya bisa fokus pada keduanya. Pohon D (yang paling kiri di taman) dikunjungi oleh keluarga Amir hanya pada waktu 7 dan oleh keluarga Badu pada waktu 2, 4, 8 dan 10.

Pohon E (tetangga paling kiri pohon D) dikunjungi oleh keluarga Amir pada waktu 6 dan 8, tetapi oleh keluarga Badu hanya pada waktu 9.

Ini Informatika!

Ilmuwan dan pemrogram komputer sering menggunakan graf (dan kemudian mereka berbicara tentang simpul dan jalur) bukannya pohon dan jalur yang menghubungkannya). Teori Graf adalah bagian yang penting dan menarik baik dalam matematika maupun informatika. Kedua ilmu ini menggunakannya dengan cara mereka sendiri untuk tujuan yang sedikit berbeda.

Hal lain yang menarik tentang tugas ini adalah representasi dari jalan-jalan di taman. Meskipun fakta bahwa beberapa pohon (simpul) ditandai dengan huruf yang sama, jalan yang dimulai dari B atau F dapat secara jelas dijelaskan oleh urutan huruf di sepanjang jalan. Ini berarti bahwa satu urutan huruf hanya menjelaskan satu langkah. Itu karena tetangga masing - masing pohon (tetangga pohon X adalah pohon yang terhubung langsung ke X melalui jalur) selalu dilabeli dengan huruf yang berbeda. Jadi kalau kita tahu di mana kita berada pada saat berjalan dan kita melihat surat berikutnya dalam representasi jalan itu tidak ada keraguan pohon mana yang harus kita kunjungi selanjutnya.

Author

2017-01-01 Andrej Blaho (Slovakia), andrej.blaho@gmail.com: Task Proposal (the date is not really known, it was sometimes in 2017)

2018-04-08 Peter Tomcsányi (Slovakia), tomcsanyi@slovanet.sk:

Refined wording, added all the yellow parts, translated to English

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/>

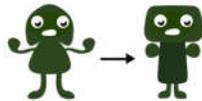


Sebuah alien mempunyai sebuah Kepala (H), sebuah Badan (B), dua Lengan (A), dan dua Kaki (L). Tubuh alien dapat ditransformasi dengan mengikuti perintah mutasi sebagai berikut. Sebuah bagian alien dapat ditransformasi lebih dari satu kali.

Perintah mutasi:

- H(C) Ubah kepala menjadi 
- H(S) Ubah kepala menjadi 
- H(T) Ubah kepala menjadi 
- B(C) Ubah Badan menjadi 
- B(S) Ubah badan menjadi 
- B(T) Ubah badan menjadi 
- A(+) Membuat Lengan lebih panjang 
- A(-) Membuat lengan lebih pendek 
- L(+) Membuat Kaki lebih panjang 
- L(-) Membuat kaki lebih pendek 

Contoh transformasi untuk H(S), B(S), A(-), L(-):



Tantangan:

Dengan mengikuti perintah mutasi secara berurutan berikut ini, bentuk tubuh alien apa yang akan terjadi?

H(T), L(+), B(T), A(+), H(C), A(-), B(C)

Pilihan Jawaban:

A	B	C	D
			

Jawaban:



Jawaban yang tepat adalah C.

Untuk setiap bagian dari alien, perintah mutasi yang terakhir akan menimpa hasil dari mutasi sebelumnya. Karena itu, hasil akhirnya adalah kepala dengan bentuk lingkaran, tubuh dengan bentuk lingkaran, lengan panjang, dan kaki panjang.

Ini Informatika!

Saat melakukan program, instruksi dijalankan secara berurutan.

Kepala, tubuh, lengan, dan kaki merupakan contoh variabel atau fungsi yang digunakan dalam suatu program.

Pengaturan bentuk: C untuk lingkaran, S untuk kotak, dan T untuk segitiga adalah seperti nilai yang ditetapkan kepada variabel atau seperti parameter untuk suatu fungsi.

Authorship

Yasemin Gülbahar, ysmnglbhr@gmail.com, Turkey

Refined by Jia-Ling Koh, jlkoh@csie.ntnu.edu.tw, Taiwan

More slight refinements by Ilya Posov, iposov@gmail.com, Russia

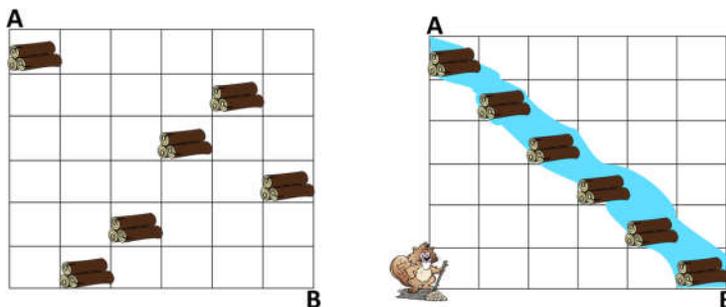
License

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/>



Insinyur Bebras si berang-berang ingin membangun bendungan untuk menahan banjir. Ia memiliki setumpuk batang pohon seperti pada **Gambar 1**. Ia ingin membuat bendungan seperti pada **Gambar 2**. Ia membutuhkan 1 jam untuk memindahkan setumpuk batang kayu pada arah vertikal, dan 2 jam untuk arah horisontal.



Tantangan:

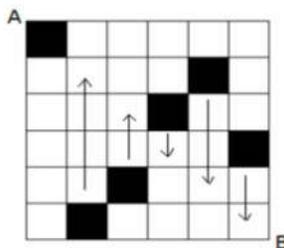
Minimal, berapa jam yang dibutuhkan oleh Insinyur Bebras untuk membangun bendungan tersebut?

Pilihan Jawaban:

- A. 11
- B. 12
- C. 14
- D. 16

Jawaban:

Jawaban yang tepat adalah B. 12.



Gambar di atas menunjukkan solusi yang membutuhkan 12 jam untuk membangun bendungan. Ini optimal, jika kita hanya menggunakan gerakan vertikal. Tapi bisakah gerakan horisontal memungkinkan solusi yang lebih baik? Ingatlah bahwa gerakan horisontal membutuhkan 2 jam. Untuk meningkatkan solusi kita, satu gerakan horisontal harus menghemat lebih dari 2 gerakan vertikal.

Perhatikan bahwa ada satu tumpukan di setiap "kolom". Jadi jika kita memindahkan tumpukan secara horisontal, kita harus memindahkan tumpukan lain secara horisontal juga, dalam arah yang berlawanan, untuk memiliki satu tumpukan di setiap kolom lagi (atau kita akan harus memindahkan tumpukan yang sama kembali ke kolom aslinya). Anda juga dapat memperhatikan bahwa dalam setiap tumpukan terletak jumlah kotak yang sama baik secara vertikal maupun horisontal dari sungai. Jadi sepertinya kita hampir tidak akan menghemat waktu dengan melakukan gerakan horisontal.

Untuk menunjukkan secara meyakinkan bahwa solusi yang diusulkan adalah optimal, kita dapat menggunakan pengamatan terakhir dan memeriksa setiap tumpukan secara individual: misalnya, cara tercepat untuk memindahkan tumpukan paling kanan ke tempat mana pun di sungai adalah dengan membuat dua gerakan ke bawah. Ini berlaku untuk semua tumpukan: solusinya kita memindahkan masing-masing ke posisi target dengan cara tercepat.

Tidak ada cara lain yang lebih cepat untuk setiap tumpukan. Inilah mengapa kita dapat mengatakan bahwa solusi yang diusulkan adalah optimal dan tidak ada cara yang lebih cepat untuk membangun bendungan.

Ini Informatika!

Tugas ini adalah tentang pengoptimalan dengan kendala. Optimasi adalah masalah yang sering harus diselesaikan di Komputer Sains (dan dalam kehidupan nyata). Untungnya, dalam kasus ini, solusinya dapat ditemukan dan diverifikasi secara mudah karena ukuran masalah, posisi tumpukan dan perbedaan waktu yang diperlukan untuk memindahkan tumpukan adalah berbeda-beda.

Misalnya, jika ada dua tumpukan dalam satu kolom, salah satunya pasti akan dipindahkan secara horizontal, dan menemukan solusi optimal dengan cepat akan menjadi jauh lebih sulit. Secara umum, mencari solusi optimal dapat memakan waktu cukup lama dan memerlukan teknik canggih seperti pemrograman dinamis.

Authorship:

Task proposal:

Selcan Kilis, selcan.kilis@giresun.edu.tr, TURKEY

Erinç Karataş, ekaratas@ankara.edu.tr, TURKEY

Workshop changes:

Andy, andrej.brodnik@fri.uni-lj.si, Slovenia

Dan Lessner, dan.lessner@centrum.cz, Czechia

License

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

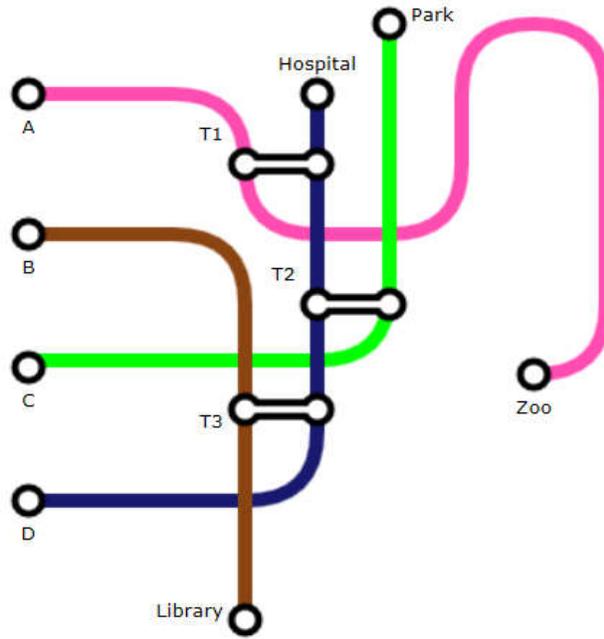
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>



Kereta Api di Kota Bebras

PENGGALANG (SMP)
I-2018-TW-02

Di kota Bebras, ada 4 jalur kereta api dimulai dari stasiun (●) A, B, C, and D. Ada juga 3 stasiun transit (⌔) T1, T2, dan T3 yang memungkinkan penumpang pindah jalur.



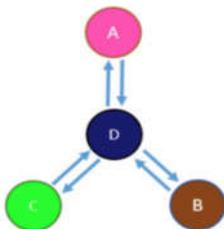
Tantangan:

Bebras Jojo akan pergi ke Zoo. Ia berganti kereta hanya sekali saja. Dari stasiun pemberangkatan mana ia berangkat?

Pilihan Jawaban:

- A. A
- B. B
- C. C
- D. D

Jawaban yang benar: D



From	To		
A	D		
B	D		
C	D		
D	A	B	C

Tidak diperlukan transfer jika John mulai di Jalur A.
 Diperlukan 2 transfer (T3 lalu T1) jika Jojo memulai pada Jalur B.
 Diperlukan 2 transfer (T2 lalu T1) jika Jojo memulai pada Jalur C.
 Hanya 1 transfer (T1) yang diperlukan jika Jojo mulai di Jalur D.

Ini Informatika!

Graf dapat digunakan untuk mewakili hubungan antar objek. Ini adalah metode untuk mewakili koneksi data dengan bantuan simpul dan garis penghubung. Graf juga memudahkan untuk menggambarkan hubungan antara hal-hal, orang atau konsep. Peta rute jalur bus atau kereta api adalah versi graf yang sangat jelas untuk dimengerti.

Ada banyak contoh kehidupan nyata dengan aplikasi graf, seperti menghubungkan dengan teman di jaringan sosial, menggunakan peta untuk menemukan rute terpendek, menunjukkan rekomendasi di situs belanja, dan sebagainya. Oleh karena itu, mentransfer situasi kehidupan nyata ke dalam graf (atau sebaliknya) adalah keterampilan penting dalam informatika.

Authorship

2018-04-08 Wei-fu Hou (Taiwan), apa@ntpc.edu.tw:

Task Revision

2018-05-08 Gerald Futschek (Austria), futschek@ocg.at: Simplification of text to be better understandable by primary children.

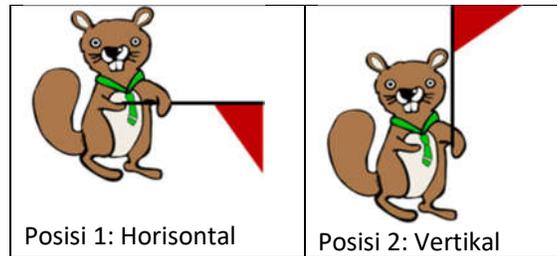
License

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

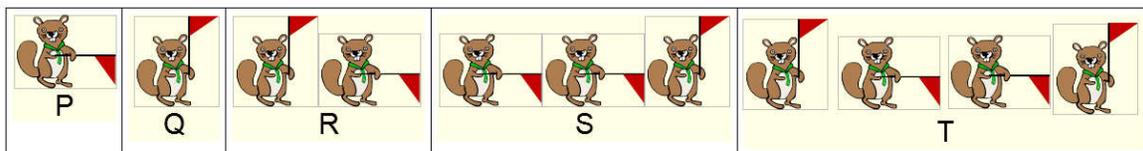
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>



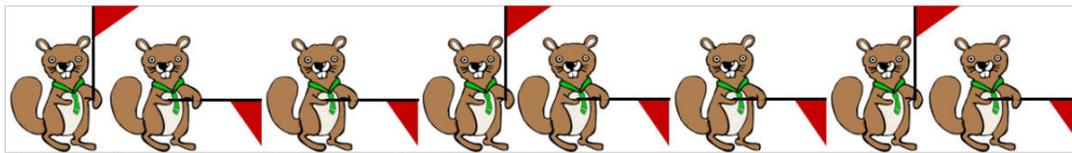
Berang-berang di desa Achi berkomunikasi dengan bendera. Bendera dapat mempunyai posisi horisontal (mendatar) atau vertikal (tegak lurus ke atas).



Mereka mengirimkan 5 huruf (P,Q,R,S,T) sesuai posisi bendera sebagai berikut:



Berang-berang Ardama mengirimkan pesan sebagai berikut:



Tantangan:

Apakah bunyi pesan tersebut?

Pilihan Jawaban:

- A. RPSP
- B. QPPTP
- C. TSQ
- D. RPQSR

Jawaban:

Jawaban yang tepat adalah B. QPPTP

Posisi horisontal dapat dikodekan sebagai 0 dan posisi vertikal sebagai 1, sehingga serangkaian posisi bendera: vertikal, horisontal, horisontal, vertikal, horisontal, horisontal, vertikal, horisontal, dapat dikodekan sebagai 10010010. Huruf P dikodekan sebagai 0, huruf Q dikodekan sebagai 1, huruf R dikodekan sebagai 10, huruf S dikodekan sebagai 001, dan huruf T dikodekan sebagai 1001. Oleh karena itu, hanya pilihan (D) yang benar. 1 (Q) 0 (P) 0 (P) 1001 (T) 0 (P).

Ini Informatika!

Mengirim pesan menggunakan bendera adalah cara yang sudah lama digunakan dalam komunikasi angkatan laut (ini disebut sistem komunikasi *semaphore*). Untuk mengirim pesan dengan cara ini, kode yang digunakan harus jelas (tidak ambigu).

Pada soal ini, kode adalah sistem yang digunakan untuk menetapkan urutan posisi bendera untuk setiap huruf. Salah satu cara untuk membuat kode yang tidak ambigu adalah membuatnya tidak tergantung kepada awalan. Ini berarti tidak ada kode huruf yang merupakan awalan dari kode huruf yang lain.

Dalam pertanyaan ini, kode tergantung awalan karena:

- kode untuk P adalah awalan dari kode untuk S,
- kode untuk Q adalah awalan kode untuk R dan kode untuk T, dan
- kode untuk R adalah awalan kode untuk T.

Kode pada soal ini ambigu, karena pesan yang sama jika dikirim dapat diterjemahkan dengan beberapa cara yang berbeda:

QPPQPPQP, QSSP, QPPTP, TSP, RPQSP, RPTP, dll.

Oleh karena itu, dapat menyebabkan pesan salah interpretasi.

Perhatikan, bahwa jika kode panjangnya sama, untuk kode yang berbeda, awalan yang sama tidak akan menyebabkan interpretasi yang berbeda.

Encoding dan Decoding adalah bagian dari domain informatika yang disebut teori informasi.

Authorship

Task Proposal: 2018-04-30 Nina Chang (Taiwan), ninachangchung@gmail.com

Task Revision:

2018-05-09 Oluchi Jennie Bamidele (Nigeria), jodamio@gmail.com

2018-05-09 Kris Coolsaet (Belgium), kris.coolsaet@ugent.be

2018-05-09 Andrej Brodnik (Slovenia)

2018-05-09 Katie Rowe (Australia), katie.rowe@csiro.au

License

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

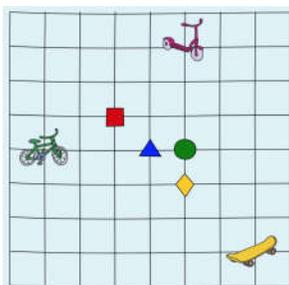
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>



Tiga sekawan Berang-berang

PENGGALANG (SMP)
I-2018-VN-03

Berang-berang Bobo () , Ali () , dan Jan () berada seperti ditunjukkan oleh kendaraannya. Ketiganya berencana untuk bertemu di suatu lokasi untuk bermain bersama. Mereka mengukur jarak dengan rumus: jumlah petak yang mendarat dan vertikal dari posisi masing-masing (hanya dapat mengikuti garis, tidak bisa menyeberangi petak).



Contoh: Pada gambar tsb, jarak Ali () dari lokasi pertemuan () adalah 6.

Tantangan:

Titik mana yang harus dipilih untuk bertemu, agar setiap berang-berang bergerak paling sedikit dari posisi masing-masing? Pilihlah Jawaban yang paling tepat.

Pilihan Jawaban:

- A. 
- B. 
- C. 
- D. 

Jawaban:

Jawaban yang tepat adalah C. 

 Salah. Total jarak dari rumah mereka ke segitiga biru adalah: $4+3+6 = 13$.

 Salah. Total jarak dari rumah mereka ke kotak merah adalah: $4+3+8 = 15$

 Benar. Jarak total dari rumah mereka ke lingkaran hijau adalah: $3+5+5 = 12$

 Salah. Total jarak dari rumah mereka ke belah ketupat kuning adalah: $4+5+4 = 13$

Ini Informatika!

Dalam Informatika, pencarian lokal dapat digunakan untuk memecahkan masalah yang dapat dirumuskan untuk menemukan solusi di antara sejumlah kriteria. Algoritme pencarian lokal berpindah dari solusi ke

solusi dalam ruang kriteria solusi (ruang pencarian) dengan menerapkan perubahan lokal, hingga solusi optimal ditemukan atau batasan waktu telah berlalu.

Untuk menemukan titik pertemuan terbaik dalam tugas ini, tambahkan semua jarak yang dicakup oleh tiga karakter untuk masing-masing berangkat-berang.

Jarak yang terpendek adalah jawaban yang benar. Menggunakan metode ini, Anda menggunakan pencarian local algoritma.

Authorship

Changes during the workshop: Gabrielė Stupurienė (Lithuania) <gabriele.stupuriene@mii.vu.lt> and

Magdalena Zaborowska-Zarach (Poland) <magdalena.zaborowska@learnetic.com>

Reviewed during the workshop: Allira Storey (Australia) <allira.storey@csiro.au>

Graphics Changes during the workshop: Vaidotas Kincius <vaidotas.kincius@gmail.com>

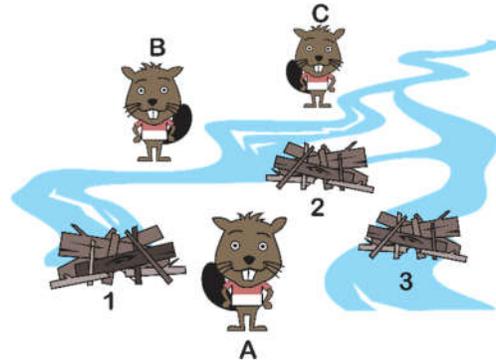
Lisence

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/>



Bapak Walikota Bebras harus memelihara 3 bendungan yang tersebar di kota setiap hari. Untuk pemeliharaan tersebut, 3 bebras A, B, C yang rumahnya tersebar akan diberi tugas pemeliharaan bendungan. Pak Walikota ingin memberikan tanggung jawab penuh, artinya satu bebras akan bertanggung jawab terhadap pemeliharaan satu bendungan.



Biaya pemeliharaan bendungan ditentukan oleh jarak yang harus ditempuh pemeliharanya. Agar biaya paling murah, Pak walikota ingin agar total jarak yang harus ditempuh oleh tiga bebras menuju ke bendungan peliharaan masing-masing minimal. Jarak dari rumah setiap berang-berang ke setiap bendungan (dalam meter) diberikan pada tabel sebagai berikut:

Dari/ke	Bendungan 1	Bendungan 2	Bendungan 3
A	125	125	130
B	125	110	110
C	150	175	175

Tantangan:

Mengacu ke tabel yang diberikan, tentukan jarak total yang minimum jika setiap bebras diberi tugas untuk memelihara 1 bendungan (dalam satuan meter).

Jawaban:

Jawaban yang benar adalah 385.

Ini Informatika!

Ini masalah kombinatorik. Diadaptasi dari Bipartite Matching. Untuk matriks M ukuran 3x3 ini cukup ide naif/serhana/bruteforce untuk memecahkannya, yakni mencoba semua kemungkinan pemetaan

$$M[A,1]+M[B,2]+M[C,3] = 125+110+175=410$$

$$M[A,1]+M[B,3]+M[C,2] = 125+110+175=410$$

$$M[A,2]+M[B,1]+M[C,3] = 125+125+175=425$$

$$M[A,2]+M[B,3]+M[C,1] = 125+110+150=385$$

$$M[A,3]+M[B,1]+M[C,2] = 130+125+175=430$$

$$M[A,3]+M[B,2]+M[C,1] = 130+110+150=390$$

Yang terkecil adalah 385.

Authorship

2018-04-09 Suryana Setiawan (Indonesia), setiawan@cs.ui.ac.id: Task Proposal
Graphics Timotius Kevin Levandi, kevin.levandi@gmail.com

Lisence

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/>



Tugas Satu Jam

PENGGALANG (SMP)
I-2018-ID-02ab-011

Berang-berang si robot dapat melakukan banyak tugas. Setiap tugas membutuhkan 1, 2, 3, atau lebih jam kerja.

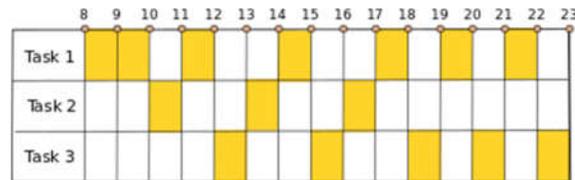
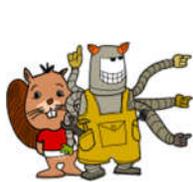
Dalam satu jam, si robot hanya dapat mengerjakan satu tugas. Pada akhir setiap jam, dia mengecek apakah ada sebuah tugas baru:

1. Jika ya, maka si robot harus mulai mengerjakan tugas baru tsb.
2. Jika tidak, si robot melanjutkan mengerjakan tugas yang paling lama tidak dikerjakannya.

Berikut ini, contoh sebuah jadwal kerja si robot dalam sehari.

- Pada pukul 8:00, ada tugas yang membutuhkan 7 jam
- Pada Pukul 10:00, datang tugas yang membutuhkan 3 jam
- Pada Pukul 12:00, datang tugas yang membutuhkan 5 jam

Pada tabel, warna kuning menunjukkan tugas tersebut sedang dikerjakan, warna putih menunjukkan tugas tersebut ditunda.



Tugas-1 selesai pada Pk 22:00, Tugas-2 selesai pada Pk 17:00, dan Tugas-3 selesai pada 23:00.

Tantangan:

Jika si robot menerima empat tugas sebagai berikut:

- Tugas-1: pada pk 8:00 membutuhkan 5 jam
- Tugas-2: pada pk 11:00 membutuhkan 3 jam
- Tugas-3: pada pk 14:00 membutuhkan 5 jam
- Tugas-4: pada pk 17:00 membutuhkan 2 jam

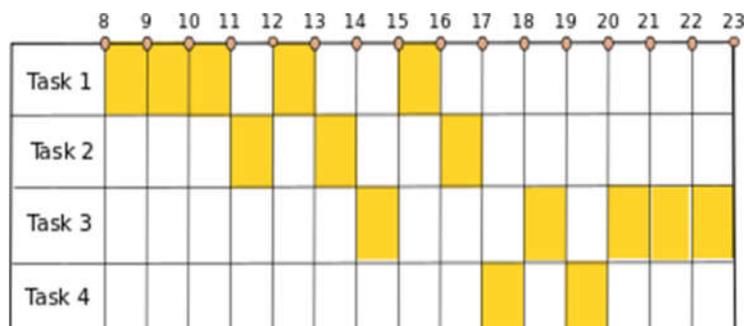
Kapan Tugas-4 akan selesai?

Isikan jawab dengan angka berupa bilangan bulat antara 0 sampai dengan 23.

Jawaban:

Jawaban yang tepat adalah 20.

Jadwalkan tugas untuk robot dengan mewarnai sel yang sesuai.



Ini Informatika!

Solusinya bisa dicari dengan hanya menggunakan simulasi grafik garis waktu sesuai dengan aturan/prosedur yang diberikan. Pertanyaan ini mencoba untuk mengekspos proses manajemen aktual berdasarkan Penjadwalan Round Robin dan penggunaan Gantt's Chart (kegiatan versus garis waktu).

Authorship

2018-04-09 Suryana Setiawan (Indonesia), setiawan@cs.ui.ac.id: Task Proposal

2018-05-09 Valentina Dagiene, Lithuania, valentina.dagiene@mii.vu.lt

2018-05-09 Thomas Ioannou, Cyprus, ioannouthomas@gmail.com

Graphics Changes during the workshop: Vaidotas Kincius <vaidotas.kincius@gmail.com>

Lisence

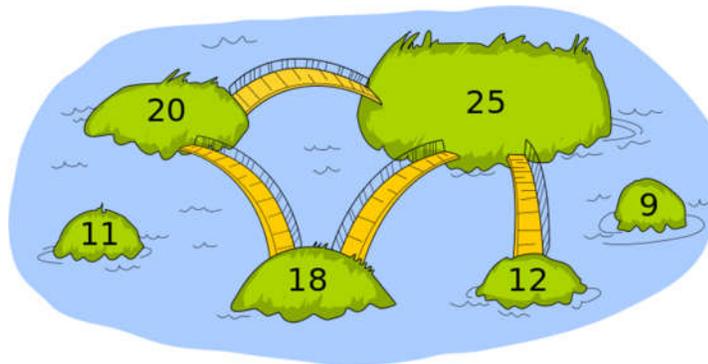
Copyright © 2018 Bebras – International Contest on Informatics and Computational Thinking.

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/>



Terdapat enam pulau yang berdekatan. Di setiap pulau tinggal sejumlah berang-berang. Angka yang tertulis menunjukkan banyaknya berang-berang pada pulau itu.

Akan dibangun jembatan-jembatan antar pulau. Sebuah jembatan yang menghubungkan dua pulau bisa dibangun jika jumlah berang-berang di kedua pulau tersebut **lebih besar dari harga suatu bilangan**. Gambar berikut menunjukkan contoh 4 jembatan yang dibangun dan jumlah penduduk setiap pulau.



Tantangan:

Berapa **harga bilangan itu** sehingga hanya empat jembatan saja yang dapat dibangun? Isikan sebuah bilangan bulat.

Jawaban:

Jawaban yang benar adalah 36.

Untuk pasangan pulau yang dihubungkan oleh jembatan, populasi dua buah yang melebihi 36 misalnya, adalah $25+12 = 37$.

Batasannya tidak boleh lebih rendah, seperti 35, karena jika tidak jumlah $25+11 = 36$ berarti lima jembatan akan dibutuhkan.

Juga tidak boleh lebih tinggi, seperti 37, karena $25+12 = 37$ tidak lebih dari 37 dan hanya tiga jembatan akan dibangun.

Ini Informatika!

Pulau dan jembatan secara alami mewakili graf, yang merupakan struktur data abstrak yang cocok untuk mewakili koneksi antar objek. Objek (pada soal ini adalah pulau) disebut simpul atau titik; dan koneksi / (pada soal ini adalah jembatan) disebut jalur.

Graf dengan n simpul mungkin memiliki $n(n-1)/2$ jalur (jika beberapa edge dikecualikan: sebenarnya, lebih dari satu jembatan mungkin menghubungkan sepasang pulau). Kadang-kadang dimungkinkan untuk menentukan jalur-jalur tanpa mendaftar semua jalur dalam graf tersebut. Satu kasus seperti itu terjadi ketika dimungkinkan untuk menentukan ambang, seperti yang telah kita lakukan untuk pulau-pulau. Pada kasus itu hanya nilai n yang harus ditentukan.

Graf yang memungkinkan ini disebut graf ambang. Tentu saja, tidak semua graf adalah ambang! Empat pulau besar akan tetap terhubung bahkan jika jembatan 25-18, misalnya, tidak akan dibangun:

graf yang sesuai tidak dapat didefinisikan melalui ambang batas.

Apakah grafiknya optimal? Itu tergantung pada definisi "optimal". Karena empat jembatan mungkin menghubungkan lima dari enam pulau, solusinya tidak optimal sehubungan dengan kriteria koneksi maksimal.

Authorship

Task Proposal: 2018-04-09 Mauro Torelli (Italy), torelli@di.unimi.it

Revision: 2018-05-09 Hongjin Yeh (Korea) hjyeh@ajou.ac.kr:

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/>



Bunda berangkat-berang sedang menghias kue ulang tahun. Ia ingin membuat kue yang berbeda-beda dengan menaruh lilin ulang tahun pada setiap kue. Bunda mempunyai dua macam lilin: merah dan kuning. Setiap kue harus dihias minimal dengan satu lilin, dan urutan warna lilin sangat penting. Misalnya lilin merah-kuning akan berbeda dengan kuning-merah walaupun keduanya terdiri dari satu lilin merah dan satu lilin kuning.

Contoh:



Bunda ingin memakai sesedikit mungkin lilin, sehingga ia mulai dengan 1 lilin, kemudian dua lilin, dan seterusnya.

Tantangan:

Jika bunda berangkat-berang harus menghias 12 kue ulang tahun, berapa minimal lilin yang dibutuhkan? Isi jawaban dengan sebuah bilangan bulat.

Jawaban:

Jawaban yang benar adalah 24.

Dua kue pertama masing-masing memiliki satu lilin (merah, kuning) = 2 lilin.

Empat kue berikutnya memiliki dua lilin (merah-kuning, merah-merah, kuning-merah, kuning-kuning) = 8 lilin.

Delapan kue berikutnya semuanya memiliki tiga lilin = 24 lilin.

Jadi solusinya adalah $2 + 8 + 24 = 34$ lilin.

Ini Informatika!

Angka biner sangat penting dalam informatika. Dalam hal ini, anda menghitung berapa banyak kombinasi berbeda yang dapat anda buat dengan "n" lilin (angka).

Dengan 1 digit anda hanya dapat membuat '0' atau '1'. Dengan 2 digit anda sudah dapat membuat 4 angka. Dengan 3 digit anda dapat membuat 8 angka, dll.

Biasanya anda akan menggunakan nomor digital untuk menghitung. Dalam pertanyaan ini kita melihat berapa banyak digit yang anda perlukan untuk menghitung hingga angka tertentu

Authorship

2018-04-08 Eljakim Schrijvers (USA), eschrijvers@eljakim.nl: Initial Task Proposal

2018-04-09 Daphne Blokhuis (USA), dblokhuis@eljakim.nl: changed the text, HTML version

2018-04-09 Andrea Schrijvers (USA), andrea@eljakim.nl: changed the story

2018-05-08 Henry Ong (Singapore), henryasmo@gmail.com: changed some text in the story to simplify the words used. 2018-05-08 Eslam Wageed (Egypt), eslamwageed@gmail.com: Added another cake to show the pattern clearly.

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/>



Berang-berang anggota Klub Komputer akan berakhir pekan dan akan tinggal dalam kamar yang besar pada sebuah wisma yang mempunyai 3 kamar. Setiap kamar dapat menampung maksimal 6 putri. Siapa yang akan tinggal dalam kamar yang sama? Setiap barang-berang menuliskan harapannya pada sebuah kartu, dengan kode:

- siapa yang disenanginya untuk sekamar dengan tanda +, dan
- siapa yang diharapkan tidak sekamar dengan tanda -.

Tantangan:

Ketua rombongan ingin agar semua anak senang. Bantulah dia menentukan kamar rombongan tersebut, sesuai dengan harapan yang ditulis pada kartu.

Untuk menentukan kamar dari setiap barang-berang, geretlah kartunya ke kotak yang disediakan.

Alina +: Lilli -:	Emma +: -:Alina	Lara +: -: Emma	Lilli +: -: Lara	Mia +: Emma, Zoe -:	Zoe +: Mia -: Alina
-------------------------	-----------------------	-----------------------	------------------------	---------------------------	---------------------------

Jawaban:

Jawaban yang benar adalah:

Alina +: Lilli -:	Lilli +: -: Lara	
Mia +: Emma, Zoe -:	Emma +: -:Alina	Zoe +: Mia -: Alina
Lara +: -: Emma		

Untuk memenuhi semua keinginan murid-murid, anda sebaiknya fokus pada keinginan positif:

1. IF (Jika) ada ruang kosong
 - a. THEN (lalu) seret kartu murid perempuan ke kamar ini,
 - b. ELSE (lain) seret kartu murid ke ruangan di mana murid tersebut tidak ada di - daftar kartu yang sudah ada di ruangan ini (dan "FAIL" GAGAL jika tidak ada kamar seperti itu)
2. Seret semua kartu gadis yang ada di + daftar kartu pertama ke ruangan yang sama. Jika tidak ada lagi kartu yang tersisa, maka anda DONE (selesai).
3. Kembali ke langkah 1.

Menerapkan prosedur di atas menghasilkan pembagian ruangan ini. Tidak ada cara lain yang akan memenuhi semua keinginan: Alina harus bersama dengan Lilli. Tapi Alina (dan karenanya juga Lilli) bisa bersama dengan Emma maupun Zoe; karenanya dia juga tidak bisa bersama dengan Mia, yang harus bersama dengan Emma dan Zoe. Lara harus pergi ke kamarnya sendiri, karena dia bisa bersama Lilli (dan karenanya Alina) maupun Emma (dan karenanya Mia dan Zoe).

Graf dengan simpul = girls dan jalur = keinginan positif (dan "counter edge" = keinginan negatif) dapat digunakan untuk menggambarkan contoh ini dan latar belakang CS lebih lanjut. Dengan kata lain, ilustrasi seperti itu mungkin bagus untuk bagian "Ini informatika!".

Seperti yang Anda lihat, ada solusi untuk pembagian kamar ini. Ini tidak selalu terjadi; misalnya, jika Alina ingin berada di ruangan yang sama dengan Lilli, tetapi Lilli tidak ingin berada di ruangan yang sama dengan Alina, kita akan menemui jalan buntu.

Ini Informatika!

Untuk menjamin semua murid senang, ketua rombongan mencari mencari cara pembagian kamar yang sesuai dengan keinginan murid-muridnya.

Bagaimana mencari solusinya? Cara sederhana adalah dengan melihat semua tugas yang bisa dibayangkan, satu demi satu, dan melihat apakah sudah memenuhi semua persyaratan. Tetapi karena kemungkinannya sangat banyak, ini akan memakan waktu cukup lama. Lebih efisien untuk mempertimbangkan persyaratan dulu dan untuk membangun tugas yang memenuhi persyaratan yang diberikan untuk mencapai solusi.

Kendala masalah kepuasan seperti masalah "tugas murid-ke-kamar" cukup terkenal di bidang informatika. Kendala mungkin muncul, seperti dalam tugas ini, ketika sumber daya perlu digunakan tanpa konflik - misalnya, ketika banyak pesawat harus dijadwalkan untuk menggunakan lepas landas atau landasan pendaratan bandara. Sistem komputer, menggunakan algoritme yang efisien, mampu memecahkan masalah seperti itu secara efisien.

Authorship

TaskProposal: 2018-03-21 Wolfgang Pohl (Germany), pohl@bwinf.de:

Revision:

Laura Ungureanu (Romania), lauungureanu@gmail.com

Tony Rene Andersen (Norway), Tony.Rene.Andersen@hfk.no

Judith Lin (Taiwan), dith007@gmail.com

Christian Datzko (Switzerland) christian.datzko@informatik-biber.ch

License

Copyright © 2018 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



Buaya Bersisik

PENGGALANG (SMP)
I-2018-AU-05

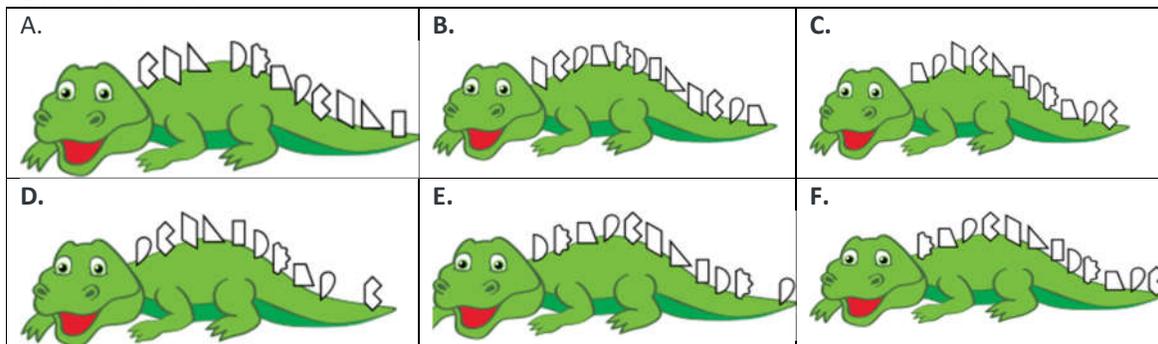
Cheri dan Charli termasuk dalam satu keluarga buaya yang sama, yaitu mempunyai sisik pada punggungnya dengan pola urutan yang sama dan berulang. Mungkin, ada sisik yang lepas namun akan tumbuh kembali yang sama pada tempatnya.



Tantangan:

Dari 5 buaya sebagai berikut, pilih buaya-buaya mana yang termasuk satu keluarga yang sama.

Pilihan Jawaban:



Jawaban:

Jawaban yang benar adalah: A, E, F

B tidak mungkin benar karena urutan pola sisiknya dari belakang ke depan.

C tidak bisa benar karena pola dan bentuk sisik-nya terbalik.

D tidak bisa benar karena ada satu sisik yang hilang di tempat di mana seharusnya setengah sisik bertemu, yang berarti kita memiliki urutan setengah silang ganda, tetapi ini tidak mungkin.

Satu-satunya jawaban yang benar bisa A, E & F karena mereka mengikuti pola bentuk sisik yang ditetapkan (seperti yang Anda lihat titik awal tidak selalu sama, dan sisik dapat hilang, tetapi mengarah pada urutan yang sama).

Ini Informatika!

Algoritma dan Programming

Algoritma adalah urutan instruksi atau seperangkat aturan untuk menyelesaikan tugas.

Data, Struktur Data dan Representasi

Data dapat mengambil banyak bentuk, misalnya, gambar, teks atau angka. Ketika kita melihat data dalam pertanyaan ini, kita sedang mencari rangkaian gambar yang akan membantu dalam memecahkan masalah. Dengan mengidentifikasi gambar-gambar ini kami dapat membuat prediksi, membuat aturan, dan menyelesaikan masalah yang lebih umum.

Authorship

(2018-02-22), task proposal, writing and significant modifications, Allira Storey, Australia, allira.storey@csiro.au

(2018-02-22), editing, Katie Rowe, Australia, katie.rowe@csiro.au

(2018-02-22), task proposal, Nicolette Venn, Australia, nicolette.venn@csiro.au

(2018-02-25), significant editing, Sarah Hobson, Australia sarah.hobson@iinet.net.au

License

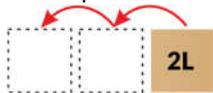
Copyright © 2017 Bebras – International Contest on Informatics and Computer Fluency.

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/>

Berang-berang Yudi senang bermain lompat petak. Terdapat 8 petak yang diberi nomor dari 1 s.d. 8. Setiap petak berisi 1 kotak yang ditandai dengan salah satu dari tiga aturan melompat.

Contoh:

- Gerakan ke kiri: Misalnya sebuah kotak ditandai "2L" berarti ia harus melompat ke kiri sebanyak 2 petak lalu menandai petak akhir lompatannya:

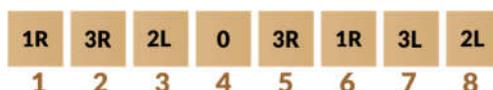


- Gerakan ke kanan: Misalnya sebuah kotak ditandai dengan "3R" berarti ia harus melompat ke kanan sebanyak 3 petak, lalu menandai petak akhir lompatannya:



- Diam. Jika aturan adalah "0", maka ia harus tetap pada tempatnya alias permainan berakhir.

Diberikan 8 petak dengan kotak-kotak sebagai berikut:



Tantangan:

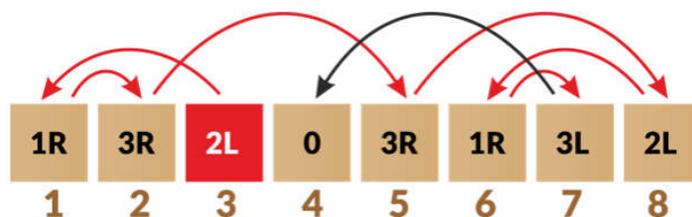
Dimulai dari petak mana kah (petak awal ini ditandai) agar kemudian setiap petak dapat ditandai tepat satu kali dan berhenti di petak dengan kotak berisi 0?

Pilihan Jawaban:

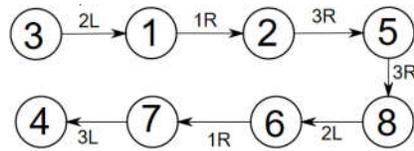
- 2
- 3
- 5
- 7
- Tidak mungkin mengunjungi semua petak.

Jawaban: 3

Berpikir mundur, kita dapat melihat bahwa petak 0 dicapai dari petak 7, yang dicapai dari petak 6, yang dicapai dari petak 8, yang dicapai oleh petak 5, yang dicapai dari petak 2, yang dicapai dari petak 1, yang dicapai dari petak 3.



Kita juga bisa menggambar ini sebagai grafik, dengan label node menjadi kolom, dan label ujungnya adalah "cara bergerak di antara kolom". Grafik ini dapat ditarik mulai dari node mana saja, dan selesai ketika semua kolom telah ditulis.



Ini Informatika!

Masalah ini mencakup beberapa petunjuk berikut: kita dapat menganggap gerakan seperti "3R" dari kolom 2 ke kolom 5 sebagai hasil dari sebuah petunjuk. Dengan kumpulan petunjuk ini, yang sebenarnya adalah grafik terarah, kita mencari simpul "head" atau "parent" dari koleksi ini.

Mengikuti serangkaian petunjuk penting dalam manajemen memori oleh sistem operasi (atau proses *garbage collection* Java) sehingga memori yang tidak lagi digunakan dapat didaur ulang dan "direklamasi" untuk digunakan oleh program lain. Kebanyakan kesalahan perangkat lunak dapat ditemukan dengan melacak kembali kalkulasi / instruksi yang bermasalah kembali ke asalnya.

Pernyataan goto (atau pernyataan lompatan dalam bahasa assembly) dapat dimodelkan oleh game ini. Pernyataan goto menunjukkan pindah ke bagian lain dari program dan lanjutkan eksekusi daripada menjalankan instruksi "selanjutnya". Dalam tugas ini, "program" adalah urutan instruksi goto, di mana satu-satunya "terminasi" instruksi adalah yang berada di posisi 4.

Authorship

Task Proposal: 2018-05-03 Troy Vasiga (Canada), troy.vasiga@uwaterloo.ca

New graphics provided by Eslam Wageed (Egypt), eslamwageed@gmail.com

Story and rule simplification by Chris Roffey (UK), chris@codingclub.co.uk

Rule correction by Rechilda Villame (Philippines) and Henry Ong (Singapore), amsliphil@yahoo.com, henrysasmo@gmail.com

License

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

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



Tiga berang-berang, masing-masing mempunyai satu meja belajar sendiri. Pada setiap meja ada 2 buku, yang anda lihat berantakan urutannya. Mereka ingin merapikannya pada beberapa kali (putaran) dan menukar buku. Pada setiap putaran, setiap berang-berang menukar dengan salah satu cara:

- Pada cara pertama, satu berang-berang menukar dua buku yang berada pada sebuah meja (Contoh A).
- Pada cara kedua, berang-berang bersebelahan menukar dua buku yang ada pada kiri-kanannya (Contoh B).



- Pada putaran pertama (dan pada setiap putaran bernomor **ganjil** berikutnya), setiap berang-berang menukar buku yang ada di mejanya (contoh A) jika keduanya tidak terurut.
- Pada putaran kedua (dan pada setiap putaran bernomor **genap** berikutnya), setiap berang-berang, kecuali di meja terkanan, menukar buku terkanannya dengan buku terkiri meja disebelah kanannya (contoh B) jika keduanya tidak terurut.
- Putaran-putaran dilakukan sampai **semua terurut**.

Tantangan:

Berapa banyaknya putaran minimal yang dibutuhkan untuk mendapatkan urutan 1, 2, 3, 4, 5, 6?

Pilihan Jawaban:

- A. 3 putaran
- B. 4 putaran
- C. 5 putaran
- D. 6 putaran

Jawaban:

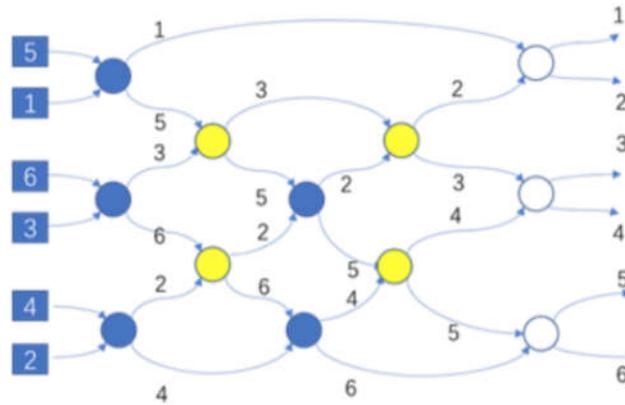
Jawaban yang tepat adalah B. 4 putaran.

Ini Informatika!

Pertanyaan ini adalah versi dari jaringan sortir paralel.

Pertama-tama mereka memulai dengan menukar buku di masing – masing meja sebagaimana ditentukan dalam pernyataan soal ini. Strategi serakah berfungsi untuk memutuskan apakah akan melakukan pertukaran atau tidak. Di babak kedua, satu-satunya hal yang mungkin adalah semua berang-berang mencoba bertukar buku antara meja. Dan kemudian di babak selanjutnya, satu-satunya hal yang mungkin adalah menukar buku di masing-masing meja, dan seterusnya.

Strategi serakah itu intuitif dan sebenarnya optimal. Dengan menjalankan strategi ini, seseorang dapat dengan mudah mencapai jawabannya. Gambar berikut menunjukkan cara kerja strategi yang disebutkan di atas dari tampilan jaringan penyortiran.

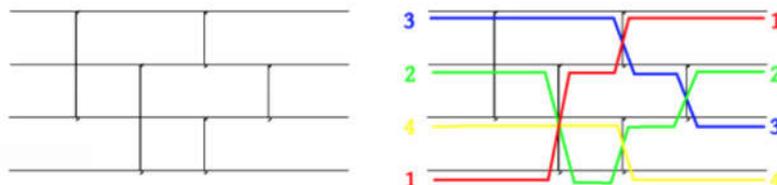


Jaringan penyortiran terdiri dari dua jenis item: pembandingan dan kabel. Kabel adalah alat perantara yang lari dari kiri ke kanan, membawa nilai (satu per kawat) yang melintasi jaringan dengan kecepatan yang sama. Setiap pembandingan menghubungkan dua kabel. Ketika sepasang nilai, bepergian melalui sepasang kabel, menemui pembandingan, pembandingan ini akan menukar nilai jika dan hanya jika nilai kawat atas lebih besar dari nilai kawat bawah.

Dalam sebuah rumus, jika kawat atas membawa x dan kawat bawah membawa y , maka setelah nilai mencapai sebuah pembandingan, masing-masing kabel membawa $x' = \min(x, y)$ dan $y' = \max(x, y)$, dan nilai mereka diurutkan.

Jaringan dari kabel dan pembandingan akan menyortir dengan benar semua input ke dalam urutan naik disebut jaringan penyortiran.

Operasi penuh jaringan penyortiran sederhana ditunjukkan di bawah ini. Sangat mudah untuk melihat mengapa jaringan penyortiran ini akan mengurutkan input dengan benar; perhatikan bahwa empat pembandingan pertama akan "menenggelamkan" nilai terbesar ke bawah dan "mengapungkan" nilai terkecil ke atas. Pembandingan terakhir cukup memilah dua kabel tengah.



Authorship

2018-03-30 Kessarapan Charoensueksa (Thailand), 1192256382@qq.com

2018-04-02 Zhang Jinbao (China), zhangjb@bnu.edu.cn

2018-05-10 Susanne Datzko, susanne@datzko.ch, Switzerland

2018-5-10 Working group 4

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/>



Ari mempunyai jalanan di halamannya yang cukup panjang. Tetangganya dapat parkir di jalan tersebut, namun hanya bisa mundur untuk keluar sebab jalannya sempit. Karena ia hanya memiliki sebuah mobil, tetangga minta izin untuk ikut parkir di jalan tersebut. Supaya yakin tidak ada yang terblokir, ia membuat tabel kapan tetangga boleh parkir, dan kapan harus pergi.

Setiap pagi, mobil yang akan pergi harus keluar sebelum mobil lainnya masuk. Seperti dapat dilihat pada tabel, tak ada yang meninggalkan jalan pada hari Senin.



Ari parkir duluan, kemudian Bob parkir setelah Ari.

Hari	Jumlah Mobil Pergi	Jumlah Mobil Masuk	Pemilik Mobil dan Urutan Mereka masuk
Senin	0	2	Ari, Bob
Selasa	1	3	Kati, Ben, Roi
Rabu	2	1	Desi
Kamis	0	2	Fina, Rosa
Jumat	3	1	Vino

Tantangan:

Mobil siapa yang akan diparkir di jalanan pada akhir hari Jumat?

Pilihan Jawaban:

- A. Bob, Vino, Desi
- B. Vino, Ari, Rosa
- C. Ari, Kati, Vino
- D. Ari, Vino, Bob

Jawaban: C. Ari, Kati, Vino.

Jika kita urutkan sepanjang minggu, berikut ini adalah urutan parkir mobil:

Akhir Senin: Ari, Bob

Akhir Selasa: Ari, Kati, Ben, Roi

Akhir Rabu: Ari, Kati, Desi

Akhir Kamis: Ari, Kati, Desi, Fina, Rosa

Akhir Jumat: Ari, Kati, Vino

Ini Informatika!

Soal ini menggunakan konsep *stack* (Tumpukan). Tumpukan adalah tipe data abstrak tempat elemen terakhir yang dimasukkan dimana elemen yang pertama akan keluar. Pengoperasian *stack* melibatkan dua fungsi: **push** (memasukkan item ke dalam stack) dan **pop** (hapus elemen dari stack). Operasi tumpukan digambarkan sebagai LIFO (terakhir masuk pertama keluar).

[https://en.wikipedia.org/wiki/Stack_\(abstract_data_type\)](https://en.wikipedia.org/wiki/Stack_(abstract_data_type))

Authorship:

2018-03-16 Pavlos Pavlikas (Cyprus), ppavlikas@gmail.com:

Task Proposal

(2018-05-10), graphics Darija Dasović Rakijašić, darija.dasovic-rakijasic@skole.hr

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/>

Ada dua stand (kios) berjualan es krim warna warni, dengan 4 warna es krim



Es krim pada stand pertama dibuat dengan mengikuti instruksi sebagai berikut:

1. Mulai dengan corong kosong.
2. Ambil warna secara sembarang (random), tambahkan 2 bulatan berwarna sama.
3. Tambah 1 bulatan dengan warna berbeda.
4. Jika tingginya sudah sesuai yang diminta, berhenti. Jika belum, kembali langkah 1.

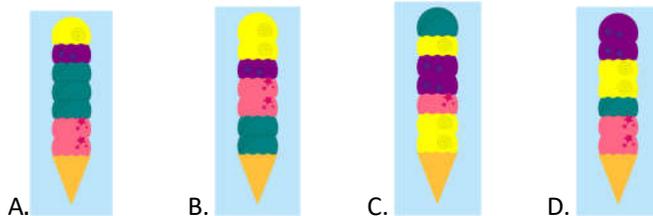
Es krim pada stand kedua tidak mengikuti instruksi tersebut.

Tantangan:

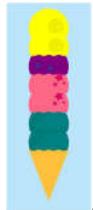
Yang mana merupakan es krim stand kedua?

Gambar yang tersedia hanya memperlihatkan beberapa susunan awal

Pilihan Jawaban:



Jawaban:



Jawaban yang tepat adalah B .

Ini adalah satu-satunya es krim yang jelas tidak mengikuti instruksi. Dimulai dengan benar dengan menempatkan dua rasa yang sama diikuti oleh salah satu rasa yang berbeda tetapi kemudian menambahkan dua sendok rasa yang berbeda ketika seharusnya menambahkan dua sendok rasa yang sama.

Jawaban A, C dan D mengikuti instruksi, setidaknya sejauh yang dapat kita lihat.

Ini Informatika!

Pola dalam kerucut es krim, atau kata-kata, atau gambar, dapat dibuat dengan daftar instruksi singkat. Mengenali pola, dan mengenali di mana pola pecah, adalah pekerjaan sehari-hari bagi para ilmuwan komputer. Terkadang pola ini berulang, misalnya, pola sederhana yang hanya terus berulang. Ini lebih mudah dikenali. Tugas ini sedikit lebih sulit, karena polanya tidak berulang.

Ada juga jebakan dalam komputer: instruksi kadang-kadang tampak seperti diikuti secara tidak sengaja. Memang, mesin kedua kadang-kadang memilih rasa secara acak dengan cara yang tampaknya mengikuti instruksi. Anda mungkin mengenali instruksi yang dilanggar. Tetapi hanya dengan pengamatan sekelintas, anda tidak akan pernah bisa yakin bahwa mereka diikuti. Untungnya, dalam tugas ini kami tahu pasti bahwa hanya satu es krim yang dihasilkan dari stand kedua.

Authorship

2018-04-09 Tom Naughton (Ireland), tomn@cs.nuim.ie

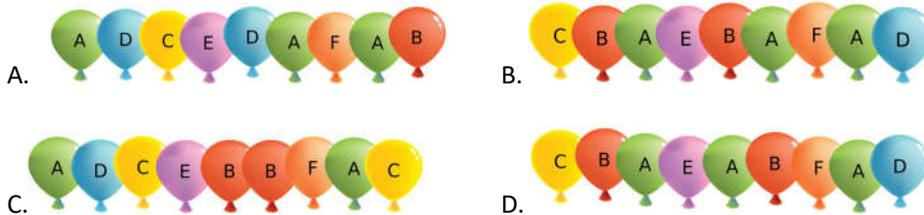
License

Copyright © 2018 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

Marni pergi ke pesta ulang tahun Mira. Dia tidak dapat melihat warna dengan baik. Warna kuning (C) akan dilihatnya sama dengan warna Hijau (A); sedangkan warna Biru (D) sama dengan Merah (B). Mira memasang sederet balon untuk menyambut tamu-tamunya.

Tantangan:

Pilih dua baris yang akan kelihatan sama bagi Marni.



Jawaban:

Jawaban yang benar adalah A dan B



Jika kita tulis simbol barisan balon di pilihan jawaban:

- A. ADCEDAFAB
- B. CBAEBAFAD
- C. ADCEBBFAC
- D. CBAEABFAD

Jika semua C kita ganti dengan A seperti yang diperintahkan di soal, maka kita akan mendapat:

- A. ADAEDAFAB
- B. ABAEBAFAD
- C. ADAEBBFAA
- D. ABAEABFAD

Jika semua D kita ganti dengan B seperti yang diperintahkan di soal, maka kita akan mendapat:

- A. ABAEBAFAB
- B. ABAEBAFAB
- C. ABAEBBFAA
- D. ABAEABFAB

Sekarang kita bisa melihat bahwa jawaban A adalah sama dengan jawaban B.

Ini Informatika!

Daftar objek adalah salah satu struktur data paling sederhana yang digunakan ilmuwan informatika, dan membandingkan daftar adalah tugas yang umum. Untuk membandingkan daftar dengan panjang yang sama, anda harus membandingkan objek yang sesuai di setiap daftar. Ketika para ilmuwan informatika menulis program untuk membandingkan benda-benda rumit, seperti cerita surat kabar atau foto orang, beberapa perbedaan harus diabaikan, dan beberapa hal berbeda harus diperlakukan sebagai "yang sama." Dalam tugas ini, kita diberikan daftar balon dan dua aturan sederhana untuk arti "yang sama".

Authorship

2018-04-09 Tom Naughton (Ireland), tomn@cs.nuim.ie:

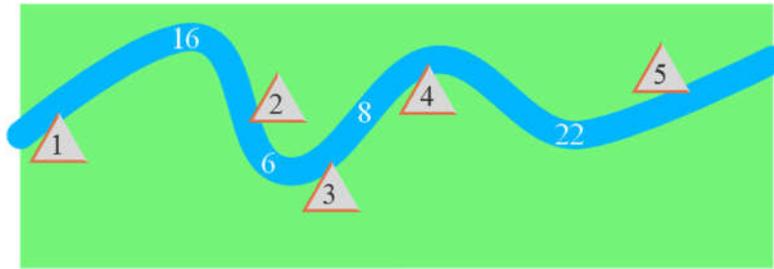
Graphics: Original graphics licensed CC0 from Jose R. Cabello (user Josethestoryteller) on pixabay.com.

Modified by Tom Naughton, tomn@cs.nuim.ie.

License

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

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



Keluarga Pak Bebras si berang-berang mempunyai 5 buah gudang makanan sepanjang sungai. Waktu yang dibutuhkan untuk pergi dari satu gudang ke gudang makanan lainnya ditunjukkan dalam gambar. Pak bebras ingin membuat rumah **di dua lokasi** pada gudang makanan tsb agar saat cuaca buruk, dari gudang manapun di antara kelima lokasi itu mereka dapat pergi ke gudang makanan terdekat. Mereka ingin membangun rumah dengan waktu penyelamatan sekecil mungkin, yaitu waktu untuk mencapai salah satu rumah paling minimal.

Tantangan:

Pilih **dua lokasi** mana yang harus dipilih untuk membangun rumah?

Jawaban:

Jawaban yang tepat adalah 5 dan 2.

Jika mereka tidak membangun rumah di lokasi makanan 5, maka berang-berang di gudang makanan 5 akan membutuhkan setidaknya 22 menit untuk mencapai sebuah gudang.

Jika gudang makanan 5 menjadi sebuah rumah, maka waktu penyelamatan menjadi kurang dari atau sama dengan 30 menit (sama dengan $16 + 6 + 8 = 30$ menit jika mereka memilih gudang makanan 5 dan 4).

Waktu maksimum akan menjadi minimal dengan memilih gudang makanan 5 dan 2: waktu penyelamatan untuk lokasi makanan 1 adalah 16 menit; waktu untuk gudang makanan 3 adalah 6, dan waktu penyelamatan untuk gudang makanan 4 adalah $8 + 6 = 14$ menit).

Ini Informatika!

Pertanyaan yang diajukan dapat diklasifikasikan di antara masalah-masalah lokasi (atau tempat) fasilitas (tidak berkapasitas). Sejumlah instalasi tertentu (pondok) dapat dibuka (di sini tanpa biaya tetap!). Dalam persoalan ini juga, kita ingin meminimalkan bukan jumlah waktu, tetapi waktu maksimum untuk sampai ke pondok terdekat.

Dalam kasus yang paling umum, masalah seperti ini NP-hard.

Authorship

2018-04-09 Lorenzo Repetto (Italy), lorenzo.repetto@istruzione.it: Task Proposal.

2018-05-09 Gary Villame (Philippines), garyvillame@gmail.com

2018-05-09 Wolfgang Pohl (Germany), pohl@bwinf.de

License

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

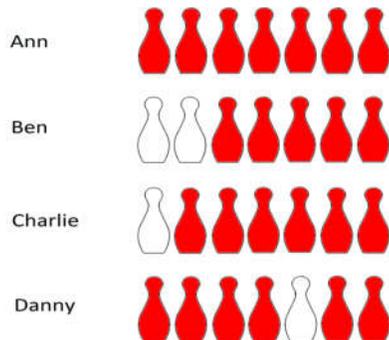
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>



Berang-berang Ani, Beni, Carla, dan Dani mempunyai pin merah dan putih. Setiap orang mempunyai 7 pin dalam satu baris, mulai dari kiri ke kanan, dengan aturan:

- pin ketiga dari kiri harus berwarna merah.
- jika sebuah pin berwarna merah, pin di sebelah kanannya harus merah.

Gambar berikut menunjukkan bagaimana pin mereka diatur.



Tantangan

Siapa yang tidak mengikuti aturan?

Pilihan jawaban:

- A. Ani
- B. Beni
- C. Carla
- D. Dani

Jawaban:

Jawaban yang benar adalah Dani.

Pin keempat berwarna merah, tetapi pin kelima berwarna putih; karena itu aturan kedua tidak diikuti. Berang-berang yang lainnya mengikuti kedua aturan dengan semua pin mereka.

Ini Informatika!

Ilmuwan informatika sering perlu memeriksa bahwa serangkaian aturan diikuti dengan ketat. Mereka bisa melakukan ini secara sistematis, dalam soal ini mereka dapat memeriksa setiap aturan terhadap setiap kelompok pin.

Ilmuwan informatika lebih suka menemukan jalan pintas yang cerdas. Dalam contoh ini mereka dapat membuat kesimpulan logis: "Jika pin berwarna merah, pin berikutnya juga harus berwarna merah" dengan kata lain: "semua pin ke kanan dari pin merah pertama harus merah", atau "pin putih mungkin hanya di sisi kiri". Proses ini membuatnya lebih mudah daripada memeriksa pin satu per satu. Dikombinasikan dengan aturan lain, anda bahkan bisa mengatakan "semua pin dari mulai pin ke tiga sampai seterusnya ke yang paling kanan harus berwarna merah".

Hal ini membuat penyelesaian masalah menjadi lebih mudah, dan anda dapat langsung memvisualisasikan solusinya.

Apa yang kami katakan di atas, pada dasarnya adalah: "Jika sesuatu selalu benar, itu pasti benar untuk semua kasus setelah itu." Jenis penalaran ini disebut "induksi" yang sangat berguna dalam informatika. Ini dapat membantu kita memecahkan masalah, seperti membuktikan apakah program itu benar atau tidak.

Author

2018-04-09 Lorenzo Repetto (Italy), lorenzo.repetto@istruzione.it: Task Proposal.

2018-05-09 Dan Lessner (Czechia), dan.lessner@centrum.cz: Changes in "it's informatics".

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/>

Yudi si berang-berang gemar pergi kemping dengan keluarganya. Ia akan membawa perlengkapan sesuai tempat kemping: Taman, Gunung, Laut atau Sungai

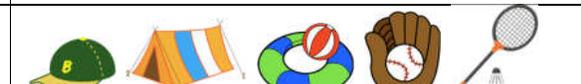
Taman		Gunung	
Laut		Sungai	

Perlengkapan Yudi disimpan di gudang. Setelah kemping, peralatan yang baru dipakai ditaruh rapi di sebelah kiri dari lainnya (menggesernya, tanpa mengubah urutan). Berikut ini foto dari gudangnya sebelum dan sesudah pergi ke taman.

Sebelum	Sesudah
	

Tantangan

Yudi pergi dua kali belakangan ini. Foto peralatannya ditunjukkan pada dua gambar berikut. Kemana saja Yudi pergi kemping 2 kali tersebut?

Sebelum	Sesudah
	

Pilihan jawaban:

- Taman, Laut.
- Taman, Sungai.
- Laut, Gunung.
- Taman, Gunung.

Jawaban:

Jawaban yang benar adalah C. Laut, Gunung.

- Setelah berkemah di taman, raket badminton dan sarung tangan baseball ada di sebelah kiri. Berkemah di laut nanti berarti tenda dan ban berenang ada di sebelah kiri.
- Setelah berkemah di taman, raket badminton dan sarung tangan baseball harus berada di sebelah kiri. Raket bulutangkis dan ban berenang kemudian akan di sebelah kiri setelah berkemah ke sungai.
- Tenda dan ban berenang harus di sebelah kiri dari segala sesuatu yang lain setelah kembali dari perjalanan berkemah ke laut. Dua item di paling kiri akan diubah ke tenda dan topi setelah perjalanan ke gunung.
- Setelah berkemah ke taman, raket bulutangkis dan sarung tangan baseball akan ditempatkan di sebelah kiri yang lainnya. Dua item ini harus tepat di bawah tenda dan topi jika ini adalah jawaban yang tepat.

Ini Informatika!

Dalam pemrograman, deretan objek biasanya disebut array.

Pada tantangan ini, anda memiliki aturan (yaitu disebut fungsi) yang mengatur ulang objek array tergantung pada variabel (tempat perkemahan).

Kemampuan untuk mengetahui input ke fungsi yang diberikan informasi yang cukup adalah keterampilan yang baik.

Jika Anda memiliki ide input mana yang menyebabkan output mana, anda akan dapat menyelesaikan masalah secara efisien.

Authorship

2018-03-31 Seung-soo Kim (South Korea), ksskbg@gmail.com: Task Proposal

2018-04-05 Dong Yoon Kim (South Korea), dykim@ajou.ac.kr: Task Modified

2018-05-08 Takeharu Ishizuka (Japan), isizukat@ishizukalab.net: Relayout / It's Informatics

2018-05-08 Jared Asuncion (Philippines), guissmo@gmail.com: Relayout / It's Informatics

018-05-09 Dimitris Mavrovouniotis (Cyprus), D.Mavrovouniotis@gmail.com: Wording, Keywords, Category

2018-05-09 Takeharu Ishizuka (Japan), isizukat@ishizukalab.net and Jared Asuncion (Philippines), guissmo@gmail.com: Relayout / It's Informatics

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/>



Para murid-murid bebras merencanakan main bola di luar. Ada beberapa pertimbangan untuk main bola di luar. Mereka hanya akan jalan-jalan di luar jika:

- hari cerah
- kecepatan angin kurang dari 20 km/jam
- lapangan bola sedang tidak dipakai kelas lain

Ramalan cuaca minggu depan adalah sebagai berikut:

	Senin	Selasa	Rabu	Kamis	Jumat
Cuaca					
Kecepatan Angin	5km/jam	24km/jam	13km/jam	7km/jam	40km/jam

Sedangkan jadwal pemakaian lapangan bola:

	Senin	Selasa	Rabu	Kamis	Jumat
Kelas	Pak Burhan	-	-	-	-

Tantangan

Pada hari apa murid-murid bebras dapat bermain bola?

Pilihan jawaban:

- Senin
- Selasa
- Rabu
- Kamis
- Jumat

Jawaban:

Jawaban yang benar adalah **Kamis**

Tiga kondisi untuk bermain sepak bola di halaman sekolah dapat diringkas dalam tabel berikut: ("OK" berarti kondisi terpenuhi, sedangkan "X" berarti kondisi tidak terpenuhi).

	Se nin	Sela sa	Ra bu	Ka mis	Ju mat
Cuaca	OK	X	X	OK	OK
Kecepatan Angin	OK	X	OK	OK	X
Lapangan Bola	X	OK	OK	OK	OK

Untuk bermain sepak bola di halaman sekolah itu, ketiga kondisi perlu di penuhi. Karena itu, satu-satunya hari yang mungkin adalah Kamis.

Ini Informatika!

Dalam database, Pengambilan Data adalah proses mengidentifikasi dan mengekstraksi data dari

database. Ini melibatkan pengambilan data dari database berdasarkan seperangkat kriteria. Pengambilan data biasanya membutuhkan penulisan dan pelaksanaan perintah pengambilan atau ekstraksi atau pertanyaan pada database. Berdasarkan permintaan yang disediakan, database mencari dan mengambil data yang diminta. Aplikasi dan perangkat lunak umumnya menggunakan berbagai pertanyaan untuk mengambil data dalam format yang berbeda. Selain sejumlah besar data, pengambilan data juga dapat mencakup pengambilan data sederhana atau lebih kecil.

Author

2018-03-31 Jin-ho Cho (South Korea), mil5079@gmail.com: Task Proposal

2018-04-05 Hong Jin Yeh (South Korea), hjyeh@ajou.ac.kr: Task Modified

2018-05-09 Khairul Anwar Mohamad Zaki (Malaysia), khairul@aidan.com.my

2018-05-09 Aliaksei Tolstikau (Belarus), tolstikov@bsu.by

changed Pollen to Wind Speed, simplified text, and improved Answer and It's Informatics.

License

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

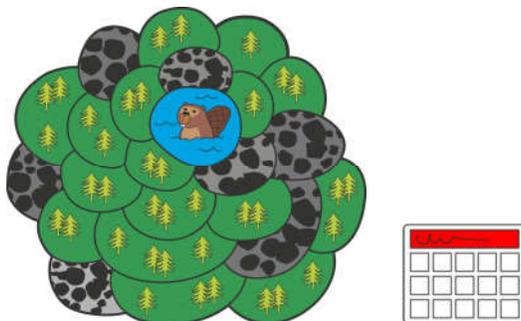
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>



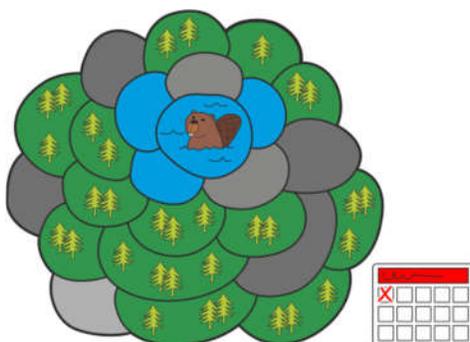
Danau Berang-berang

PENGGALANG (SMP)
I-2018-LT-02a

Berang-berang hidup dalam sebuah lembah dikelilingi gunung. Dalam lembah, ada danau. Danau dikelilingi lapangan-lapangan yang berisi pohon-pohon atau batu-batu.



Setiap hari, para berang-berang akan mengalirkan air ke lapangan yang ada pohonnya yang bersebelahan dengan danau, atau yang bersebelahan yang sudah dialiri air. Misalnya, pada hari pertama, 3 lapangan sebagai berikut dialiri:



Tantangan

Setelah berapa hari semua area yang berpohon diisi air?

Pilihan jawaban:

- A. 3
- B. 4
- C. 5
- D. 6

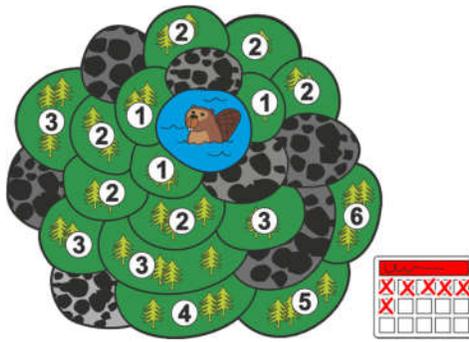
Jawaban:

Jawaban yang benar adalah 6.

Angka-angka dalam gambar di bawah ini menunjukkan daerah yang diairi setelah jumlah hari yang sesuai.

Area di sebelah danau ditandai dengan 1. Kemudian semua area yang tidak ditanami oleh pohon di sebelah area yang ditandai dengan 1, ditandai dengan 2. Proses ini dilanjutkan sampai semua area dengan pohon ditandai.

Seperti yang bisa kita lihat di atas, area terakhir dengan pohon akan banjir setelah 6 hari.



Ini Informatika!

Ilmuwan informatika mempelajari berbagai jenis algoritma. Algoritma adalah urutan langkah-langkah untuk memecahkan sebuah masalah. Soal ini menunjukkan algoritma *wavefront* yang digunakan untuk menandai area yang dibagi menjadi sel-sel kecil.

Proses ini menggunakan pendekatan pencarian luas pertama (BFS). Algoritma BFS dimulai pada sel yang disebut “akar” (*root*). Kemudian, pada setiap langkah, setiap sel di sebelah sel yang dikunjungi sebelumnya dikunjungi. Pada soal ini, sel mewakili area yang mengandung pohon, dan mengunjungi sel sama dengan mengairi area.

Kalau dilakukan dengan teliti, BFS mengunjungi setiap sel di area yang akan ditandai hanya sekali. Ini dilakukan secara terorganisir.

Bahkan, seperti yang terlihat dalam jawaban, sel-sel dapat diberi nomor saat dikunjungi. Angka-angka ini menggambarkan seberapa jauh jarak setiap sel dari “akar” (*root*). Artinya, algoritma BFS dapat digunakan untuk menemukan jalur terpendek dari “akar” (*root*) ke semua sel lainnya.

<http://www.cs.tufts.edu/comp/150IR/labs/wavefront.html>

http://en.wikipedia.org/wiki/Breadth-first_search

Authorship

Valentina Dagiene, Lithuania: valentina.dagiene@mii.vu.lt

2018-05-09 J.P. Prettì (Canada), jpretti@uwaterloo.ca

2018-05-09 Dimitris Mavrovouniotis (Cyprus), D.Mavrovouniotis@gmail.com

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/>



Twist and Turn

Tom adalah seorang anak berusia 10 tahun yang tinggal di sebuah kota penuh belokan dan putaran. Hari itu, telpon ibunya ketinggalan di rumah, dan ia meminta Tom untuk mengantarkannya ke kantornya. Ibu memberikan sebuah peta agar Tom tidak tersesat. Peta tersebut digambarkan sebagai sebuah persegi dengan petak-petak yang dinomori 1 s.d. 6. Dan panah dari A ke F. Peta tersebut juga mengandung tanda arah yang dapat ditempuh Tom.

	Tom hanya dapat turun
	Tom dapat ke bawah atau ke kanan
	Tom dapat ke kiri atau ke bawah
	Tom dapat ke atas atau ke kanan
	Tom menemui halangan sehingga tidak dapat bergerak lagi

Tom mulai dari pojok kiri atas (A1).

	1	2	3	4	5	6
A						
B						
C						
D						
E						
F						

Tantangan:

Rute mana yang dapat ditempuh oleh Tom untuk sampai ke kantor ibunya (F6)?

Pilihan Jawaban:

- A. A1 B1 B2 B3 C3 D3 E3 E4 F4 F5 F6
- B. A1 B1 B2 B3 B4 C4 D4 D5 D6 E6 F6
- C. A1 B1 B2 B3 C3 D3 E3 F3 F4 F5 F6
- D. A1 B1 B2 B3 C3 D3 D4 D5 D6 E6 F6

Jawaban:

Jawaban yang tepat adalah C. A1 B1 B2 B3 C3 D3 E3 F3 F4 F5 F6

Jawaban A tidak benar karena di E5 Tom akan menemukan halangan.

Jawaban B tidak benar karena Tom tidak dapat berpindah dari B4 ke C4. Dari B4 ia hanya bisa pindah ke B3 atau ke A4.

Jawaban D tidak benar karena Tom tidak dapat beralih dari D3 ke D4. Dari D4 ia hanya bisa pindah ke D2 atau E3.

Jadi jawaban yang benar adalah C.

Ini Informatika!

Struktur yang digunakan dalam tugas ini adalah matriks yang digunakan untuk menggambarkan peta dan array satu dimensi yang digunakan untuk menggambarkan arah bergerak pada pilihan jawaban. Masing-masing elemen array terdiri dari dua komponen: karakter dan angka.

Untuk menyelesaikan tugas ini, anda harus membaca setiap elemen array jawaban dan melacak setiap elemen matriks. Meskipun masalahnya nampak sederhana, pembacaan elemen array dan elemen matriks secara simultan cukup sulit karena kondisi yang diberlakukan oleh nilai elemen matriks.

Anda dapat menemukan solusi dengan menerapkan pencarian *Depth First Search* (DFS) yang juga disebut *backtracking*. Cara lain adalah mulai dari ujung jalan dan mencoba mencapai titik awal. Kedua metode ini dapat digabungkan untuk menghasilkan sebuah metode yang efisien jika hanya ada beberapa solusi (jalur).

Authorship

2018-04-09 Laura Ungureanu (Romania), lauungureanu@gmail.com

2018-04-09 Corina Vint(Romania), corina.vint@yahoo.com

2018-05-10 Anton Chukhnov (Russia), septembreange@gmail.com, reduced the table to 6x6.

2018-05-10 Maciej M. Sysło (Poland) syslo@ii.uni.wroc.pl

New Graphics is done by Vaidotas Kincius at Bebras Workshop 2018

License

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

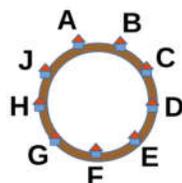
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>



Mengunjungi Rumah Teman

PENGGALANG (SMP)
I-2018-SK-04

Yugo si berang-berang tinggal di sebuah desa yang isinya hanya 9 rumah, ditata secara melingkar mengelilingi danau. Pintu masuk rumah diberi jarak 10 meter antara satu sama lain. Pada hari Lebaran, Yugo ingin mengunjungi tetangganya. Yugo mulai dari rumahnya (H), berjalan ke salah satu arah dan akan mengunjungi sebuah rumah jika ia memutuskan untuk berhenti. Setelah selesai berkunjung, ia akan berjalan lagi (bisa berbeda arah), dan mengunjungi salah satu rumah lain yang belum dikunjungi. Setiap kali ia mengunjungi sebuah rumah, ia mencatat jarak dari rumah ia berangkat sampai dengan rumah yang dikunjungi sebelumnya.



Tantangan:

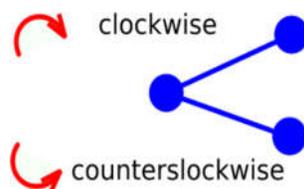
Berangkat dari rumah (H), Yugo berakhir di rumah F, dan mencatat jarak yang ditempuh: 20 10 20 40. Pilih 3 rumah tetangga yang dikunjungi Yugo sebelum sampai ke F.

Pilihan Jawaban:

Tuliskan huruf dari rumah-rumah yang dikunjungi oleh Yugo

Jawaban:

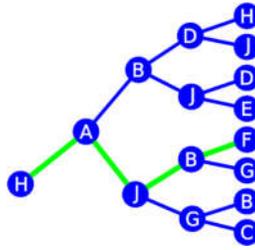
Jawaban yang benar adalah A, B dan J.



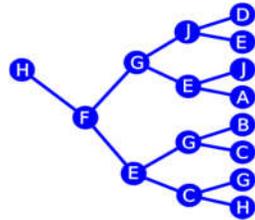
Gambar ini membuktikan bahwa Yugo bisa mencapai rumah F sesuai dengan jumlah yang dicatat dengan mengunjungi rumah-rumah ini. Pertama-tama, Yugo berjalan searah jarum jam ke A, kemudian dia berjalan berlawanan dengan arah jarum jam ke J, di bagian ke tiga dia berjalan searah jarum jam ke B dan akhirnya searah jarum jam ke F.

Kita harus membuktikan bahwa tidak ada kemungkinan cara berjalan yang lain. Perhatikan bahwa ada 4 jarak jalan dan masing-masing jarak bisa dijalani maju (searah jarum jam), atau mundur, (berlawanan arah jarum jam), jadi ada $2 * 2 * 2 * 2 = 16$ jalur yang mungkin dijelaskan oleh angka-angka ini. 2 diagram pohon di bawah ini membuktikan bahwa selain dari 16 cara ini tidak ada lagi kemungkinan Yugo untuk mencapai rumah F.

1. Jika Yugo memulai dengan berjalan 20 meter searah jarum jam, ia tiba di rumah A. Diagram pohon berikut menunjukkan semua kemungkinan, baik searah jarum jam maupun berlawanan arah jarum jam dari rumah A.



2. Jika Yugo memulai dengan berjalan 20meter berlawanan arah jarum jam, ia bisa tiba di rumah F. Semua kemungkinan jalur dari F ditunjukkan di bawah ini.



Rumah terakhir ditampilkan di ujung kanan kedua grafik dan dapat dilihat bahwa rumah F hanya dikunjungi sekali dan itu dengan melalui A, J dan B. Jadi itu adalah satu-satunya solusi.

Ini Informatika!

Ilmuwan dan pemrogram komputer seringkali perlu mewakili beberapa objek dunia nyata (dalam hal ini jalur yang diambil saat mengunjungi tetangga) di komputer sebagai data. Mereka perlu menemukan representasi yang baik.

Representasi adalah baik jika kita dapat dengan mudah menggunakannya untuk merekonstruksi semua properti objek dunia nyata yang akan kita butuhkan untuk proses yang ingin kita lakukan.

Representasi jalan kaki Yugo dengan merekam hanya titik awal, titik akhir dan jarak berjalan (tetapi bukan arah), mungkin representasi yang baik jika kita perlu tahu panjang total

Jalan Yugo (mudah dihitung sebagai $20 + 10 + 20 + 40 = 90$). Tapi itu tidak bagus jika kita ingin tahu rumah mana saja yang dikunjungi Yugo. Anda dapat melihat bahwa untuk mengetahuinya kita perlu memecahkan semacam teka-teki dan dalam beberapa kasus mungkin ada beberapa kemungkinan solusi. Representasi yang lebih baik untuk contoh ini, misalnya, untuk merekam jarak dan arah (plus atau minus seperti dalam penjelasan).

Author

2017-01-01 Andrej Blaho (Slovakia), andrej.blaho@gmail.com:

Task Proposal (the date is not really known, it was sometimes in 2017)

2018-04-08 Peter Tomcsányi (Slovakia), tomcsanyi@slovanet.sk:

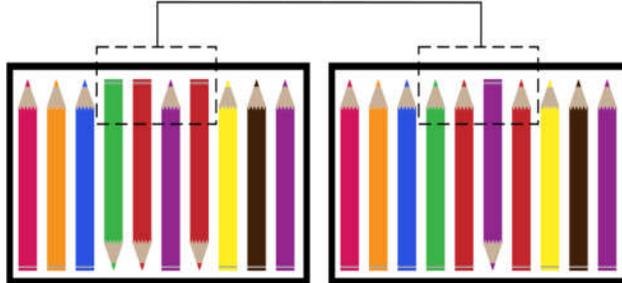
Refined wording, added all the yellow parts, translated to English

License

Copyright © 2018 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

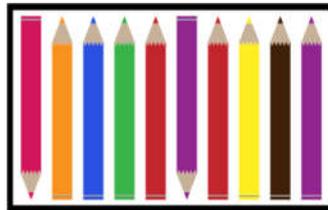


Ade mempunyai sekotak krayon. Beberapa krayon ada yang menghadap ke atas, ada yang menghadap ke bawah. Menurut Ade, sekotak krayon akan “indah dipandang” kalau semua krayon di dalamnya menghadap ke arah yang sama. Pada satu langkah, ia dapat membalikkan sederet krayon dalam satu baris; setelah melakukan ini maka semua krayon yang semula menghadap ke atas akan menghadap ke bawah dan sebaliknya yang semula menghadap ke bawah akan menghadap ke atas, seperti pada gambar.



Tantangan:

Untuk posisi krayon sebagai berikut, berapa langkah minimum harus dilakukan agar Ade mempunyai kotak krayon yang “indah dipandang”? Jawab dengan mengetikkan sebuah bilangan bulat.

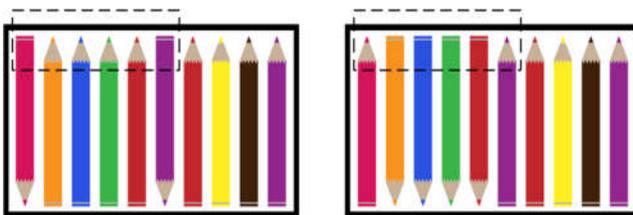


Jawaban:

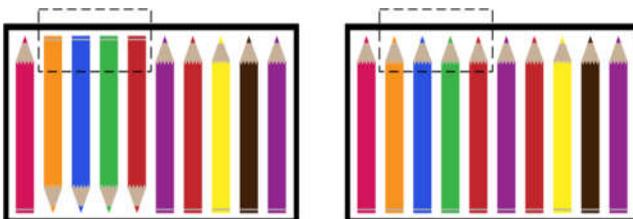
Jawaban yang tepat adalah 2.

Kita tidak dapat menyelesaikan masalah dengan hanya membalik satu urutan, karena 2 krayon yang mengarah ke bawah terpisah, mereka tidak bersebelahan. Namun untuk mendapatkan kotak yang indah dipandang, bisa dilakukan dalam 2 langkah:

Langkah pertama: balikkan krayon ke-1, ke-2, ke-3, ke-4, ke-5 dan ke-6.



Langkah kedua: balikkan krayon ke-2, ke-3, ke-4 dan ke-5.



Ini Informatika!

Memecahkan masalah dengan langkah minimum adalah salah satu keterampilan terpenting dalam ilmu komputer dan kehidupan. Seorang programmer yang baik selalu ingin mencari solusi terbaik dari suatu masalah. Untuk pertanyaan ini, solusi terbaik hanya ditemukan dalam 2 langkah, tetapi dalam kasus yang lebih sulit mungkin tidak mudah menemukannya. Sebuah program yang baik dapat membantu.

Author

2018-04-09 Zsuzsa Pluhár (HU), pluharzs@caesar.elte.hu:

Task Proposal based on idea: <https://www.codechef.com/problems/ADACRA>

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/>