



Tim Olimpiade Komputer Indonesia

TANTANGAN
Bebras Indonesia 2018
BAHAN BELAJAR COMPUTATIONAL THINKING

Tingkat SMA



<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 otomatisasi 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:

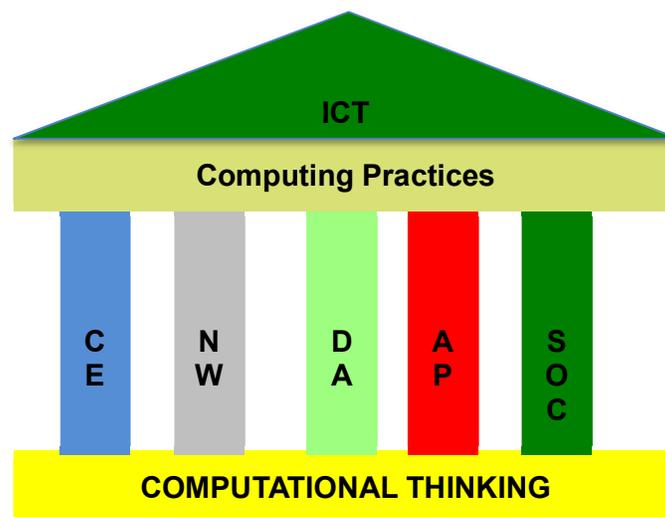
- Memformulasikan persoalan sehingga dapat menentukan solusinya, baik yang akan diselesaikan dengan bantuan komputer, atau *tools* lainnya
- Mengorganisasikan dan menganalisis data secara logis;
- Merepresentasikan data melalui abstraksi dalam bentuk model, dan melakukan simulasi;
- Melakukan otomatisasi 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 memraktekkannya, 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 Seymour Papert dari MIT. Siswa diajak belajar dengan mencoba menjawab tantangan. Jadi, tantangan Bebras bukan lomba sekedar untuk menang tetapi yang lebih penting adalah untuk belajar berpikir dan menyelesaikan persoalan. Kepada peserta yang meraih peringkat tinggi, akan diberikan sertifikat.

Tujuan Tantangan Bebras:

- Memotivasi siswa 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 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 SMA Tantangan Bebras 2018

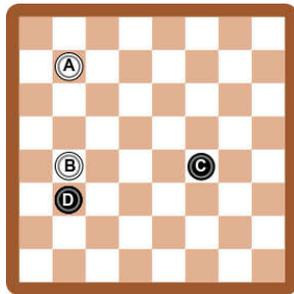


DAFTAR SOAL

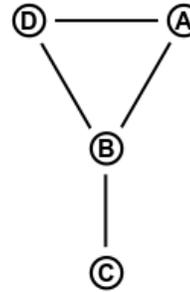
- Baris dan Kolom (halaman 10)
- Jadwal Gladi Resik (halaman 12)
- Robot Pemotong (halaman 14)
- Kata Terpanjang (halaman 16)
- Hadiah Pertemanan (halaman 18)
- Infinite Ice Cream (halaman 21)
- Harga Terlambat Bangun (halaman 23)
- Jalan Tol (halaman 25)
- Titik Utama Wifi (halaman 27)
- Peta Harta Karun (halaman 29)
- Petak dan Kotak (halaman 32)
- Antrian Mobil (halaman 34)
- Tempat Perlindungan (halaman 36)
- Dua Berang-Berang Pekerja (halaman 37)
- Lift Pengangkut Barang (halaman 39)
- Medical Labs (halaman 41)
- Siapa Berbohong? (halaman 43)
- Twist and Turn (halaman 45)
- Kotak Krayon (halaman 47)
- Bendungan 2 (halaman 49)
- Tugas Satu Jam (halaman 51)
- Tiga Sekawan Berang-berang (halaman 53)



Pada Gambar-1 berikut, ada sebuah papan permainan dengan 4 buah koin, yang digambarkan sebagai diagram pada Gambar-2.



Gambar-1

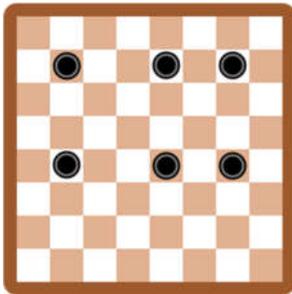


Gambar-2

Pada gambar-2, setiap koin digambarkan sebagai sebuah lingkaran. Jika dua buah koin berada pada baris dan kolom yang sama pada papan permainan, maka gambarkan sebuah garis yang menghubungkan kedua buah koin tersebut. Tidak ada garis lain dalam diagram, selain yang menghubungkan dua buah koin seperti di atas. Huruf yang dituliskan pada setiap koin akan membantu untuk memeriksa apakah diagram benar.

Tantangan:

Untuk gambar papan permainan dengan 6 koin sebagai berikut yang memang tidak kelihatan hurufnya, diagram mana yang benar?



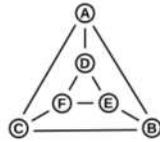
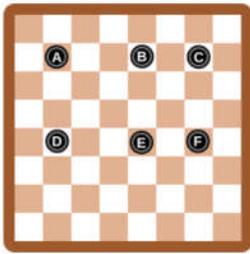
Pilihan Jawaban:

A	B	C	D

Jawaban:

Jawaban yang benar adalah **B**

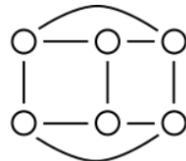
Anda dapat memeriksa dengan cepat dari gambar papan yang sudah di mana koin diberi huruf sebagai berikut:



Pada papan tersebut, setiap koin memiliki dua koin lainnya di baris yang sama dan satu koin lainnya di kolom yang sama. Oleh karena itu, dalam diagram, setiap lingkaran harus terhubung ke $2 + 1 = 3$ lingkaran lainnya.

Jawaban D memiliki 7 lingkaran, satu terlalu banyak, jadi jelas salah.

Mungkin Anda berpikir diagram A itu benar? Kelihatannya, diagram A sangat mirip dengan letak koin di papan. Namun, koin paling kiri dan paling kanan dalam diagram hanya terhubung dua koin lainnya, sehingga diagram tersebut tidak benar. Untuk membuat diagram A menjadi benar, Anda dapat menambahkan dua garis, seperti pada gambar berikut:



Ini Informatika!

Dalam informatika, diagram seperti ini sering digunakan untuk merepresentasikan informasi penting dari suatu masalah. Diagram seperti itu disebut graf. Lingkaran dalam diagram disebut simpul dari graf.

Dalam sebuah graf, yang penting untuk diketahui adalah apakah dua simpul terhubung atau tidak. Posisi simpul digambar, tidak penting. Graf yang sama sering dapat digambar dengan cara yang berbeda (posisi simpulnya berbeda), seperti ilustrasi di atas: jawaban B dan gambar terakhir dalam penjelasan adalah diagram yang benar dan mewakili graf yang sama.

Tergantung pada informasi apa yang anda butuhkan untuk menyelesaikan masalah, graf dapat menjadi representasi yang berguna untuk memecahkan masalah informatika.

Jika misalnya Anda perlu tahu apakah koin putih berada di baris atau kolom yang sama dengan koin hitam, maka graf bukan representasi yang baik: warna koin tidak terwakili dalam simpul. Jika anda perlu tahu apakah masih ada koin yang tersisa di papan sehingga permainan masih bisa dimenangkan, maka grafnya 'berlebihan': anda hanya perlu melacak berapa banyak koin yang tersisa di papan setiap saat, dan tidak perlu mencatat koin mana yang berada di baris atau kolom yang sama dengan koin lain.

Di sisi lain, graf adalah representasi yang baik untuk menjawab pertanyaan seperti "Berapa jumlah minimal koin yang harus anda hapus sehingga tidak ada bagian dalam kolom baris yang sama dengan bagian lainnya?".

Menemukan representasi yang tepat untuk suatu masalah adalah salah satu tantangan yang dihadapi pemrogram komputer atau ilmuwan informatika dalam pekerjaan mereka.

Authorship

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

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

2018-05-08 Troy Vasiga (CA), troy.vasiga@uwaterloo.ca: Edited the task to create alternative pictures with just black pieces, in order to lessen the difficulty slightly.

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/).



Sekolah Bebras akan mengadakan pertunjukan menari, dengan penari berpasangan. Ada 6 penari yaitu : Ana, Budi, Cinta, Dori, Evi, Fani.

Mereka akan menari berpasangan :

1. Ana - Budi
2. Evi - Dori
3. Ana - Evi
4. Budi - Cinta
5. Dori - Ana
6. Fani - Budi
7. Cinta - Evi
8. Budi – Dori
9. Dori - Fani
10. Fani - Evi

Pelatih ingin menjadwalkan gladi resik untuk suatu tarian berantai. Dalam sebuah tarian berantai, urutan tarian ditentukan sedemikian rupa sehingga dari satu tarian ke tarian berikutnya, salah satu dari pasangan penari akan tetap tinggal di panggung untuk pertunjukan berikutnya. Selain itu, ada aturan bahwa seorang penari tak boleh dijadwal menari 3 kali berturut-turut, sebab akan kelelahan.

Contoh: saat Ana dan Evi menari, salah satu alternatif berikutnya adalah Cinta dan Evi. Setelah itu, Evi tidak dapat menari lagi.

Pertanyaan:

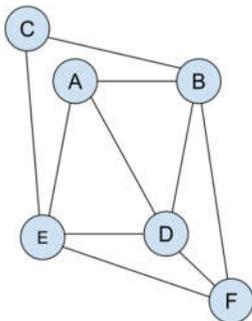
Penari mana yang tak boleh dijadwalkan pada tarian pertama karena akan menyebabkan tidak mungkin membuat pertunjukan tarian berantai?

Pilihan Jawaban:

- A. Ana
- B. Cinta
- C. Evi
- D. Dori
- E. Budi
- F. Fani

Jawaban:

Jawaban yang benar adalah B. Cinta.



Perhatikan diagram di samping.

Dalam diagram ini penari diwakili oleh lingkaran. Dua lingkaran terhubung oleh sebuah garis ketika dua penari harus menari berpasangan. Aturan jadwal latihan digambarkan oleh garis di diagram ini. Perhatikan mulai dari sebuah lingkaran, kita dapat mengikuti garis dari lingkaran tersebut ke lingkaran lain sampai kita melacak setiap garis penghubung tepat satu kali.

Karena saat melacak garis ini kita harus meninggalkan setiap lingkaran yang kita lewati, ini berarti bahwa setiap lingkaran harus memiliki garis penghubung dengan jumlah genap, kecuali yang pertama dan yang terakhir di jalur. Jadi, lingkaran pertama di jalan harus A atau F.

Dengan kata lain, jadwal apa pun yang dipilih, pasangan penari pertama harus Ana atau Fani. Cinta adalah satu-satunya penari yang tidak memiliki pasangan dengan keduanya. Jadi dia tidak akan pernah bisa menjadi bagian dari pasangan pertama.

Kalau kita perhatikan setiap garis, kita harus mengambil arah dari penari yang sudah ada di latihan sebelumnya ke penari yang akan berada di latihan berikutnya. Karena satu penari tidak akan menari tiga kali berturut-turut, setiap langkah di sebuah garis akan menuju ke suatu lingkaran yang berbeda dalam diagram. Garis penghubung dalam diagram berkorespondensi dengan satu pasang penari.

Inilah Informatika!

Diagram penjelasan jawaban di atas adalah graf. Jalurnya (digambarkan sebagai garis) adalah jalur Euler. Graf adalah sebuah cara untuk membuat model dari suatu situasi yang terdiri dari titik atau simpul. Dalam soal ini simpul adalah para penari. Setiap garis menghubungkan dua penari. Garis dalam graf ini mewakili pasangan penari. Dalam soal ini anda akan menggunakan semua pasangan, sedemikian rupa sehingga setiap garis hanya digunakan satu kali. Euler membuktikan bahwa hal ini hanya mungkin jika anda memulai dan berhenti di sebuah simpul dengan jumlah hubungan ganjil.

Mewakili informasi menggunakan graf dapat sangat membantu untuk melihat struktur di balik suatu masalah. Perencanaan rute dan penjadwalan adalah contoh lain aplikasi penggunaan graf.

Authorship

2018-04-09 Veerle Fack (BE), veerle.fack@ugent.be

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/).



Angelo si berang-berang mendapat hadiah sebuah robot yang dapat membantunya untuk menanam pohon untuk membuat kebun. Robot mengerti perintah sebagai berikut:

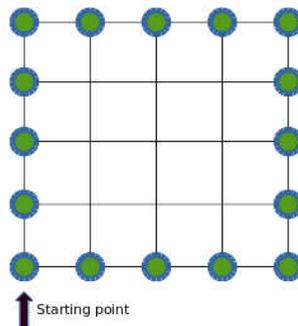
Perintah	Arti
Start	Hidupkan robot
Maju (X)	Robot maju X meter
Mundur (X)	Robot mundur X meter
KeKiri (X)	Robot berputar ke kiri
KeKanan (X)	Robot berputar ke kanan
Tanam	Robot menanam Pohon
Ulangi X (instruksi)	Robot mengulangi instruksi dalam kurung sebanyak X kali
Stop	Matikan Robot

Ada 16 lokasi yang harus ditanami pohon pada sebuah lapangan berbentuk persegi. Sisi lapangan ukurannya 8 meter dan setiap pohon harus ditanam dengan jarak 2 meter.

Robot berada pada posisi pojok kiri bawah dengan arah seperti ditunjukkan oleh panah. Pada awalnya, robot pada status mati dan setelah selesai menanam pohon, harus dimatikan. Setelah sebuah pohon ditanam, robot dapat melanjutkan gerakan tanpa halangan sepanjang garis-garis pada gambar.

Tantangan:

Program yang mana yang akan membuat robot menanam semua pohon sepanjang sisi lapangan seperti ditunjukkan pada gambar?



Pilihan Jawaban:

- A. Start
Ulangi 4{
 Ulangi 4{Tanam; Maju(2)},
 KeKanan(90)}
Stop
- B. Start
Ulangi 4{
 Ulangi 4{ Tanam, Maju (2)},
 KeKiri(90)}
Stop
- C. Start
Ulangi 4{
 Ulangi 4{ Maju (2), Tanam },
 KeKiri (90)}
Stop

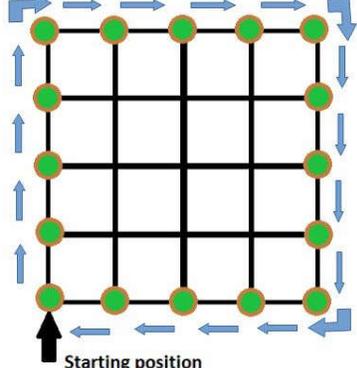
```

D. Start
  Ulangi 4{
    Ulangi 4{ Maju (1), Tanam },
    KeKanan (90)}
  Stop

```

Jawaban:

Jawaban yang tepat adalah

<pre> Start Ulangi 4{ Ulangi 4{Tanam; Maju(2)}, KeKanan(90)} Stop </pre>	
--	--

Robot akan mulai dari **Ulangi 4 {Tanam, Maju (2)}** jadi dia akan menanam 4 pohon pertama di atasnya. Setelah itu, robot akan berbelok ke kanan (90°).

Robot akan mengulangi perintah ini 3 kali sesuai dengan Ulangi 4.

Jawaban B dan C tidak akan berfungsi karena mereka membelokkan robot ke kiri, menjauh dari lapangan.

Jawaban D tidak akan berfungsi karena Maju (1) tidak memajukan robot pada jarak yang tepat untuk mencapai pohon selanjutnya.

Ini Informatika!

Tantangan ini adalah tentang pemahaman algoritma seperti yang diterapkan dalam program komputer. Pada tantangan ini, algoritma melibatkan urutan langkah yang harus dilakukan oleh robot untuk menyelesaikan tugas yang diinginkan. Berpikir tentang urutan langkah merupakan keterampilan penting untuk pemrograman robot.

Perhatikan juga bahwa ada pengulangan di dalam pengulangan lainnya, yang disebut pengulangan bersarang. Biasanya, pengulangan digunakan untuk mengulang suatu perintah.

Ketika Anda memiliki array sederhana atau hanya satu dimensi, anda bisa menggunakan pengulangan sederhana. Namun, jika bekerja dalam dua dimensi (seperti dalam sebuah matriks), anda akan memerlukan pengulangan bersarang.

Authorship

2018-03-30 Demetris Hadjipantelis (Cyprus), hpdemetr@cytanet.com.cy

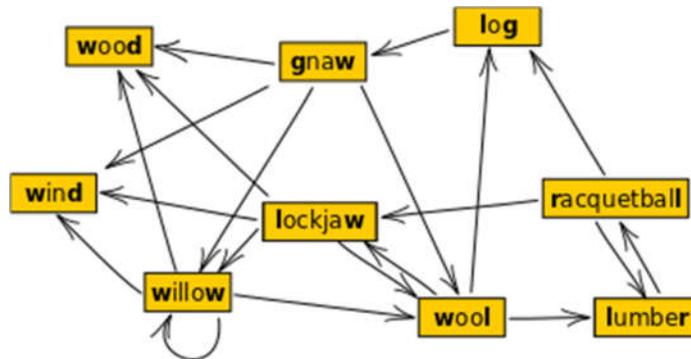
License

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/>



Sekumpulan berang-berang sedang bermain “rantai kata” dalam bahasa Inggris. Salah satu berang-berang memulai dengan mengucapkan sebuah kata. Berang-berang lainnya harus mengucapkan sebuah kata lain yang dimulai dengan huruf terakhir dari kata sebelumnya sampai tak ada kata yang dapat diucapkan. Permainan akan diulang lagi mulai dari sebuah kata lain, dan seterusnya. Sejujurnya, kelompok bermain tersebut belum mengenal banyak kata-kata bahasa Inggris, sehingga rantai kata yang dapat diucapkan terbatas kepada kata-kata sebagai berikut:



Tantangan:

Berapa banyak kata yang maksimum dapat disebutkan dalam sebuah permainan?

Jawaban:

Jawaban yang tepat adalah 8.

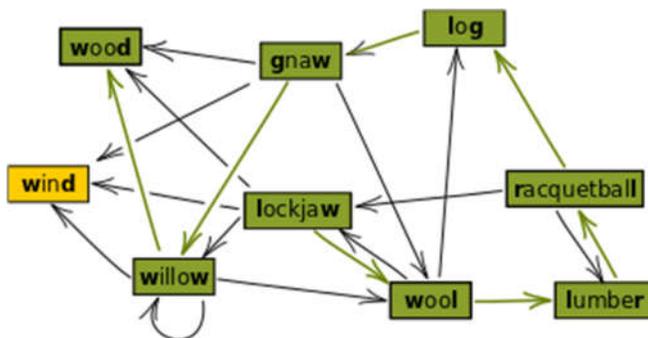
Berang-berang dapat menggunakan paling banyak 8 kata di tiap permainan. Salah satu contohnya adalah: **lockjaw-wool-lumber-racquetball-log-gnaw-willow-wood**.

Bisakah kamu menemukan permainan lain dengan panjang yang sama?

Menurut contoh ini, berarti kita yakin bahwa satu permainan dapat menggunakan setidaknya 8 kata. Tapi kita belum tahu apakah mungkin untuk menggunakan semua 9 kata (9 kata) tersebut.

Sekarang perhatikan kata kayu (**Wood**) dan angin (**Wind**). Tidak ada kata-kata yang dimulai dengan d, jadi jika kata-kata ini akan digunakan, itu harus menjadi kata terakhir dari permainan. Tetapi kita tidak mungkin punya dua kata sebagai kata terakhir.

Oleh karena itu, kita tidak dapat menggunakan semua 9 kata tersebut dalam satu permainan.



Ini Informatika!

Tugas ini mengharuskan Anda untuk memahami seperangkat aturan (cara memainkan rantai kata, https://en.wikipedia.org/wiki/Word_chain), representasi data yang jelas dan praktis (grafik), dan

kemudian menggunakannya untuk menemukan solusi optimal dalam sistem yang diberikan (kosa kata). Semua ini cara tipikal ilmuwan komputer menghabiskan hari-hari mereka.

Dengan pengetahuan ekstra tentang informatika, anda bisa mengenali masalah yang diberikan saat mencari jalur terpanjang dalam graf berarah. Ini adalah contoh problem yang terkenal (https://en.wikipedia.org/wiki/Generalized_geography) dan terkait erat dengan “*traveling salesman problem*” (persoalan penjual berkeliling) yang terkenal (<https://www.youtube.com/watch?v=SC5CX8drAtU>).

Informatika memberikan cara untuk menemukan permainan terpanjang dengan kosa kata yang lebih banyak (bayangkan berapa ribu kosa kata yang anda ketahui!) yang tentunya sangat sulit untuk dicari, dan membutuhkan waktu sangat lama bahkan dengan algoritma terbaik yang kita miliki hingga saat ini. Mungkinkah anda bisa mendapatkan yang lebih baik?

Authorship

2018-04-09 Dan Lessner (Czechia), dan.lessner@centrum.cz

License

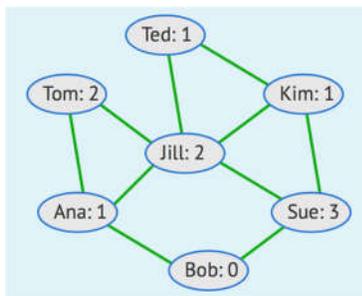
Copyright © 2018 Bebras – International Contest on Informatics and Computational Thinking.

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/>



Hadiah Pertemanan

Diagram sebagai berikut menunjukkan hubungan pertemanan di antara sekumpulan berang-berang. Dua berang-berang berteman jika namanya dihubungkan oleh sebuah garis. Mereka merencanakan sebuah pesta. Untuk setiap pasangan teman, satu di antara mereka harus membawa hadiah untuk teman lainnya. Angka menunjukkan berapa hadiah yang dapat dibelinya.



Tantangan:

Tentukan untuk setiap pasangan teman, siapa yang akan membeli hadiah untuk temannya. Setiap berang-berang tidak dapat membeli hadiah lebih dari jumlah yang harus dibelinya.

Nama di kolom pertama adalah nama berang-berang yang membeli hadiah (Pemberi), pilih berang-berang yang akan menerima hadiah tersebut di kolom Penerima.

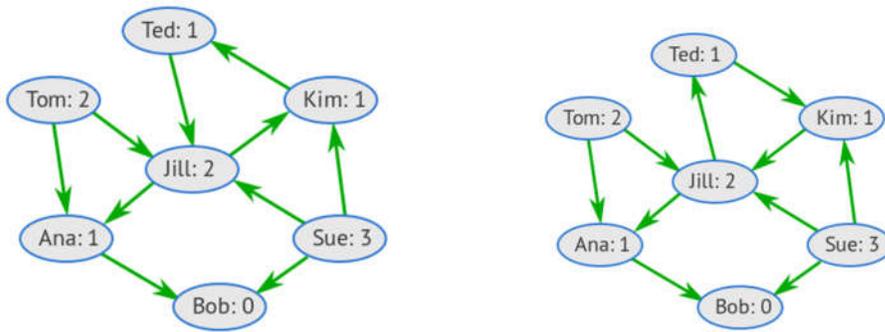
Pemberi	Penerima	Penerima	Penerima
Jill			
Sue			
Ted			
Tom			
Ana			
Kim			

Jawaban:

Pemberi	Penerima	Penerima	Penerima
Jill	Kim	Ana	
Sue	Jill	Bob	Kim
Ted	Jill		
Tom	Jill	Ana	
Ana	Bob		
Kim	Ted		

Kita memakai hubungan arah panah untuk menunjukkan pemberi dan penerima. Jika panah menunjuk dari teman A ke teman B, A akan membeli hadiah untuk B. Ada dua cara yang mungkin untuk memilih arah memberi dan mematuhi semua aturan pemberian.

Untuk memilih arah dengan benar, kita mulai dengan Bob. Karena dia tidak punya uang, dia hanya bisa menerima hadiah dari teman-temannya Ana dan Sue. Ana sudah menghabiskan semua uangnya, sehingga dia hanya bisa menerima hadiah dari teman-temannya Jill dan Tom. Artinya, untuk pertemanan ini, tidak ada pilihan.



Sekarang kita tampaknya memiliki banyak pilihan. Tetapi ada pilihan kritis: Ted dan Kim keduanya dapat membeli 1 hadiah saja. Untuk pasangan ini, kita perlu memutuskan siapa yang akan membeli hadiah untuk yang lain.

- Jika kita memilih Kim untuk membeli hadiah untuk Ted, maka Kim perlu menerima hadiah dari teman-temannya Jill dan Sue. Jadi hadiah yang bisa dibeli Jill telah mencapai angka tertinggi. Jill harus menerima hadiah dari Ted, Tom, dan Sue. Solusi ini ditunjukkan di sebelah kiri.
- Jika kita memilih Ted untuk membeli hadiah untuk Kim, maka Ted perlu menerima hadiah dari Jill. Sekali lagi, Jill telah memberikan semua hadiahnya dan harus menerima hadiah dari Tom, Kim dan Sue. Sekali lagi, Sue juga perlu membeli hadiah untuk Kim. Solusi ini ditampilkan di sebelah kanan.

Artinya, setiap pilihan untuk Ted dan Kim tidak meninggalkan pilihan untuk persahabatan yang tersisa. Karena hanya ada dua pilihan untuk Ted dan Kim, hanya ada dua jawaban yang benar. Kemudian kita bisa melihat apakah Jim atau Ted saling memberi hadiah sesuai dengan persyaratan.

Ini Informatika!

Anak-anak dan pertemanan di antara mereka membentuk jaringan pertemanan dengan simpul (anak-anak) dan garis penghubung (pertemanan). Ini mengingatkan kita pada jejaring sosial terkenal yang digunakan banyak orang. Tetapi ada perbedaan penting di antara jaringan-jaringan ini, beberapa di antaranya, ada "persahabatan" timbal balik (misalnya: garis penghubung tanpa arah), seperti dalam tantangan. Di jaringan lain, ada "pengikut", sehingga garis penghubung memiliki arah. Anda dapat mengikuti beberapa "VIP", tetapi VIP mungkin tidak mengikuti Anda juga.

Dalam soal ini, garis penghubung pertemanan dua arah harus dibuat sebagai garis penghubung "pemberian", dan berbeda dari "mengikuti", ada batasan atau kapasitas untuk memberi. Kita ingin melihat sebanyak mungkin hadiah berpindah dari satu anak ke anak lain (semoga ada satu hadiah dalam setiap pertemanan), tanpa melebihi kapasitas siapapun. Ini mirip dengan masalah yang dikenal luas dalam informatika, yaitu diharapkan kapasitas garis penghubung terbatas, tetapi dalam batas-batas ini kita ingin memiliki sebanyak mungkin aliran melalui penghubung tersebut. Perbedaannya adalah bahwa dalam tantangan ini, node memiliki kapasitas, sedangkan di "masalah aliran jaringan" garis penghubunglah yang memiliki kapasitas. Namun, tidak sulit untuk mengubah masalah ini menjadi masalah aliran jaringan nyata. Untuk menyelesaikan soal ini, bahkan dengan banyaknya anak dan pertemanan; kita bisa menggunakan algoritma yang efisien untuk masalah aliran jaringan terkenal di bidang informatika. Ini adalah cara khas bagaimana masalah dipecahkan dalam informatika, yaitu dengan "mengubah"nya menjadi masalah lain yang solusinya terkenal.

Authorship

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

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

2018-05-08 Troy Vasiga (CA), troy.vasiga@uwaterloo.ca: Edited the task to create alternative pictures with just black pieces, in order to lessen the difficulty slightly.

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) penjual es krim warna warni, dengan 4 warna es krim



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

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

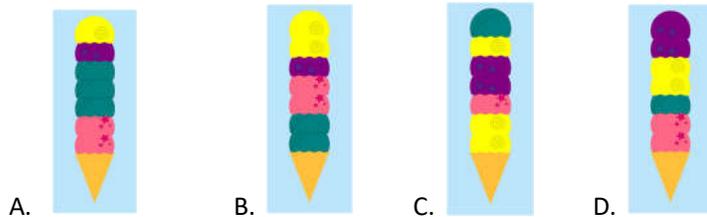
Es krim pada stand kedua tidak mengikuti instruksi tersebut.

Tantangan:

Yang mana merupakan es krim stand kedua?

Gambar yang tersedia hanya memperlihatkan beberapa susunan awal

Pilihan Jawaban:



Jawaban:



Jawaban yang tepat adalah B.

Es Krim B 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 salah karena mereka 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 bagian-bagian pola, adalah pekerjaan sehari-hari para ilmuwan informatika. Terkadang, sebuah pola berulang, misalnya, pola sederhana yang hanya terus berulang. Ini lebih mudah dikenali. Tantangan 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 selintas, anda tidak akan pernah bisa yakin bahwa mereka diikuti. Untungnya, dalam soal 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 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/>



Harga Terlambat Bangun

PENEGAK (SMA)
I-2018-IR-03-X-001-5

Bob Bekerja di stasiun Pusat Kota, dan jam kerja dimulai pukul 8:00. Bob akan didenda jika datang terlambat. Untuk setiap 15 menit terlambat, ia harus membayar denda sebesar Rp. 10.000,-. Misalnya jika ia tiba sebelum pukul 8:15 maka ia tidak didenda. Jika ia datang pukul 8:20 maka ia akan didenda Rp. 10.000,-. Pagi ini, Bob ketiduran dan tiba di stasiun keberangkatan pukul 8:08.



Tabel berikut menunjukkan tabel keberangkatan berbagai kereta menuju Stasiun Pusat Kota dan harga tiketnya:

Kereta	Jadwal	Waktu tempuh ke stasiun Pusat	Harga tiket
Biasa	Mulai Pk. 6:00 Setiap 05 menit	40 menit	Rp. 5000,-
Wira-Wiri	Mulai pk 6:00 Setiap 10 menit	30 menit	Rp 10.000,-
Cepat	Mulai Pk 7:00 Setiap 15 menit	20 menit	Rp 15.000,-
Ekspres	Mulai Pk 7:00 Setiap 20 menit	12 menit	Rp 20.000,-

Tantangan: kereta mana yang harus diambil Bob agar walaupun terlambat, tetap paling “murah” dendanya?

Pilihan Jawaban:

- A. Wira-Wiri
- B. Biasa
- C. Cepat
- D. Ekspres

Jawaban:

Jawaban yang tepat adalah A. Wira-Wiri.

Wira-Wiri adalah jalur yang paling hemat biaya untuk Bob karena:

- Jika dia mengambil jalur Biasa, dia akan tiba di tempat kerja pada pukul 08:50, sehingga akan dikenakan total denda Rp. 35.000,- yaitu Rp. 5000,- untuk tiket ditambah Rp. 30.000,- (3 * Rp. 10.000,-) denda untuk keterlambatan.
- Jika dia mengambil jalur Wira-Wiri, dia akan tiba di tempat kerja pada pukul 08:40, sehingga akan dikenakan denda Rp. 30.000,- yaitu Rp 10.000,- untuk tiket ditambah denda keterlambatan Rp 20.000,- (2 * Rp 10.000,-).
- Jika dia mengambil jalur Cepat, dia akan tiba di tempat kerja pada pukul 08:35, jadi akan dikenakan denda Rp. 35.000,- yaitu Rp 15.000,- untuk tiket ditambah Rp 20.000,- (2 * Rp 10.000,-) denda keterlambatan.
- Jika dia menggunakan jalur Ekspres, dia akan tiba di tempat kerja pada pukul 08:32, jadi biayanya Rp 40.000,- yaitu Rp 20.000,- untuk tiket ditambah Rp 20.000,- (2 * Rp 10.000,-) denda keterlambatan.

Ini Informatika!

Soal ini memperkenalkan konsep Optimasi. Optimasi dengan cara yang paling sederhana berarti memilih elemen terbaik (di sini pilihan yang paling hemat dendanya) dari beberapa alternatif yang tersedia (di sini berbagai jalur kereta) berdasarkan beberapa kriteria (di sini waktu dan biaya).

Masalah optimisasi umumnya terdiri dari memaksimalkan atau meminimalkan nilai suatu fungsi dengan secara sistematis memilih nilai input dan menghitung nilai fungsi tersebut.

Authorship

2018-04-09 Hamed Mohebbi (Iran), hmdmohebbi@gmail.com: Task Proposal

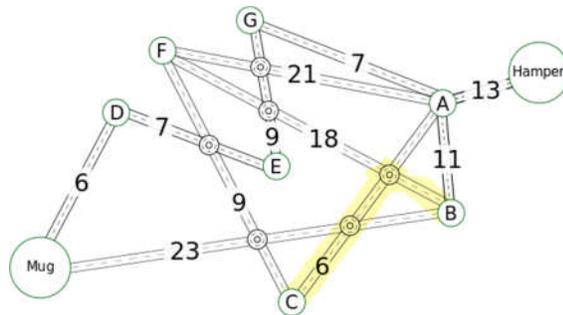
License

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/>



Bobi si berang-berang memutuskan untuk bepergian dari Hamper ke Mug. Pada peta, lingkaran adalah sebuah kota dan sebuah garis adalah sebuah ruas jalan tol dua arah yang menghubungkan kedua kota. Huruf menunjukkan nama kota. Angka menunjukkan biaya yang harus dibayar saat masuk jalan tol yang menghubungkan dua kota tersebut. Mobil dapat berpindah arah saat ada sebuah persimpangan tapi tetap harus membayar penuh jalan yang dimasukinya. Misalnya untuk bepergian dari kota B ke C, dapat dipilih jalan sehingga membayar $24 = 18+6$.



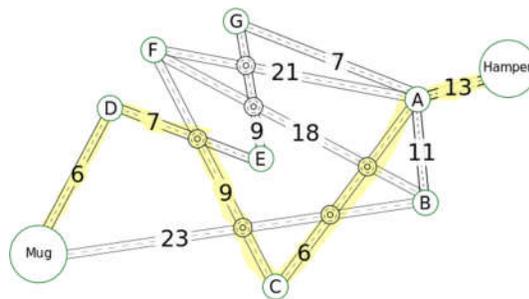
Tantangan:

Berapa biaya paling murah dari Hamper ke Mug? Isikan sebuah bilangan bulat

Jawaban:

Jawaban yang tepat adalah 41.

Jalan termurah ditunjukkan di bawah ini:



Untuk membuktikan bahwa jalur ini optimal, seseorang dapat membandingkan pilihan dengan cara berikut.

Jalan termurah yang kita temui biayanya 41, mari kita coba mencari jalan yang lebih murah. Pada awalnya, jelas bahwa jalan itu melalui jalan A – Hamper dengan biaya 13, jadi orang harus menemukan jalur dari A ke Mug yang lebih murah dari $41 - 13 = 28$.

Kedua, jelas bahwa kita harus datang ke Mug bukan melalui jalan B – Mug, karena itu sudah 23, dan tidak mungkin menemukan jalur dari B ke A yang lebih murah dari $28 - 23 = 5$.

Jadi, kami datang ke Mug di jalan D – Mug dengan biaya tol 6, dan dengan demikian kita harus menemukan jalur dari A ke D yang lebih murah dari $28 - 6 = 22$.

Jalur dari A ke D tidak boleh memiliki lebih dari tiga jalan, karena jalan termurah memiliki biaya 6, dan dengan demikian 4 jalan akan memberikan biaya setidaknya $6 * 4 = 24$.

Jalan pertama yang mungkin kita lihat untuk keluar dari A adalah A – B. Ini mengarah ke solusi yang jauh lebih mahal, karena kita akan menggunakan jalan tol yang biayanya sangat mahal, 23 atau 18.

Argumen yang sama berlaku untuk jalan A – F, karena jalan ini sangat mahal.

Jika kita menggunakan jalan A – G, maka kita akan membutuhkan biaya setidaknya $7 + 9 + 7 = 23$ yang lebih besar dari 22.

Dan akhirnya, jika kita melewati jalan A – C, kita dapat mencapai tol optimal.

Ini Informatika!

Dalam teori graf, masalah jalur terpendek adalah tentang menemukan jalur antara dua simpul (atau node) dalam grafik sedemikian rupa sehingga jumlah bobot dari tepi penyusunannya diminimalkan. Titik graf sesuai dengan persimpangan dan ujungnya sesuai dengan ruas jalan, masing-masing diukur oleh baik jaraknya, atau tolnya, atau beban lalu lintas, atau waktu, dll.

Masalah menemukan jalur terpendek antara dua persimpangan di peta jalan dengan jenis tol seperti itu yang tidak tergantung pada jarak yang dilalui di sepanjang jalan dapat disederhanakan menjadi masalah jalur terpendek klasik dengan merepresentasikan persimpangan menggunakan simpul dan jalur penghubung tambahan. Dan masalah jalur yang terpendek klasik dapat diselesaikan misalnya dengan algoritma Dijkstra.

https://en.wikipedia.org/wiki/Dijkstra's_algorithm

Authorship

2018-04-09 Ali Abedini (Iran), amin1382math@gmail.com: Task Proposal

Ilya Posov (Russia), iposov@gmail.com: Working group 9

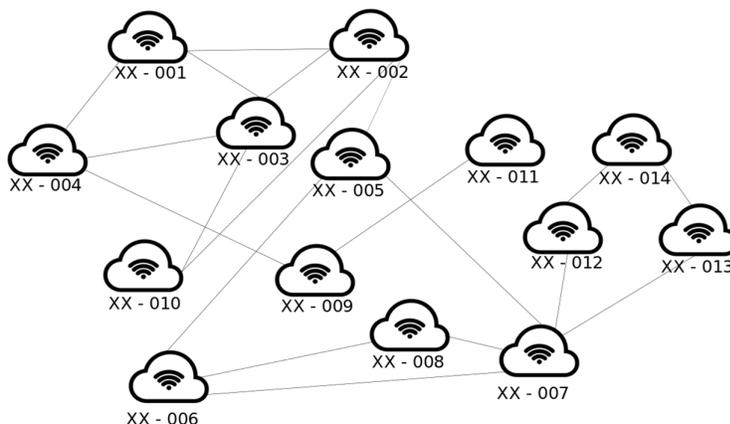
Georgios Fessakis (Greece) gfesakis@aegean.gr: Working group 9

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/>

Jaringan lokal rumah Bebras dilengkapi dengan 14 titik akses (*Access Point*) ke Wifi. Pada jaringan ini, beberapa *Access Point* disebut Titik Kunci (*Key Point*), yang jika rusak akan menyebabkan Titik Akses lain tidak berfungsi. Misalnya, Titik Akses XX-009 adalah sebuah Titik Kunci: jika XX-009 rusak, maka XX-011 tidak dapat mengakses jaringan lagi.



Tantangan:

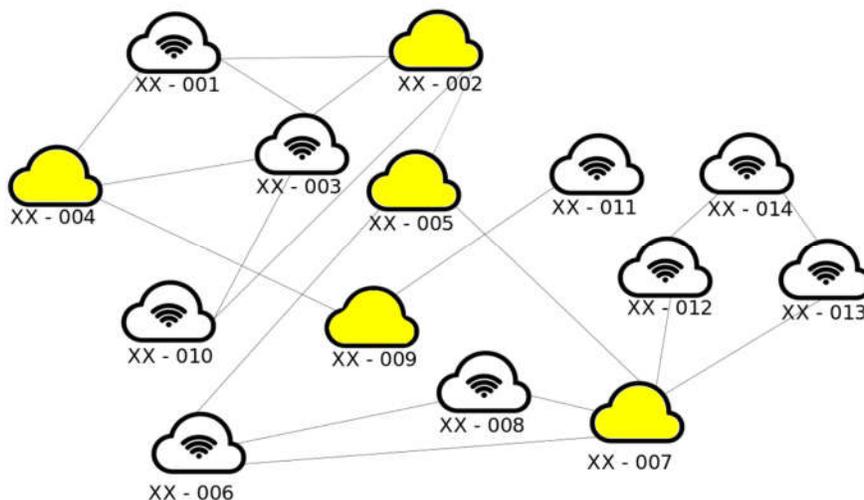
Titik Akses mana saja yang merupakan Titik Kunci? **Jawaban benar bisa lebih dari satu.**

Pilihan Jawaban:

- a) XX-13
- b) XX-02
- c) XX-14
- d) XX-07
- e) XX-06
- f) XX-03
- g) XX-08
- h) XX-01
- i) XX-12
- j) XX-09
- k) XX-04
- l) XX-05
- m) XX-11
- n) XX-10

Jawaban:

Jawaban yang benar: XX-02, XX-07, XX-09, XX-04, XX-05



Ini Informatika!

Praktik umum dalam Informatika adalah menggunakan graf sebagai struktur data untuk mewakili jaringan yang merupakan sekumpulan keterhubungan. Pada soal ini, tugasnya terdiri dari menemukan simpul yang memisahkan graf menjadi paling sedikit 2 komponen terkait dalam graf yang terhubung, yang disebut titik artikulasi. Untuk menentukan titik artikulasi dalam graf yang tidak berarah, kita bisa menggunakan algoritma pencarian kedalaman-pertama yang dimodifikasi untuk menyimpan informasi tambahan untuk setiap node.

Mari kita anggap bahwa T adalah graf. Kemudian, simpul v adalah titik artikulasi jika:

- 1) v adalah akar T dan v memiliki dua anak atau lebih; atau
- 2) v bukan akar T dan memiliki anak u di T sehingga tidak ada simpul dalam sub- graf yang didominasi oleh u yang terhubung dengan leluhur v melalui jalur penghubung sesudahnya (anak-anaknya tidak dapat mencapai tingkat yang lebih tinggi pada graf dengan menggunakan jalur lain).

Authorship

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

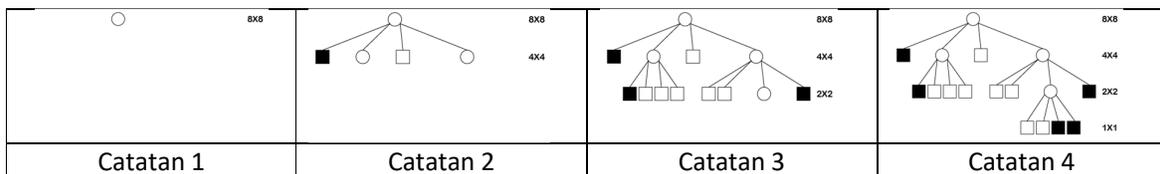
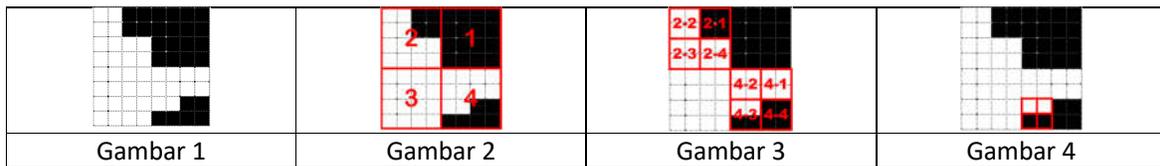
License

Copyright © 2018 Bebas – 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/>

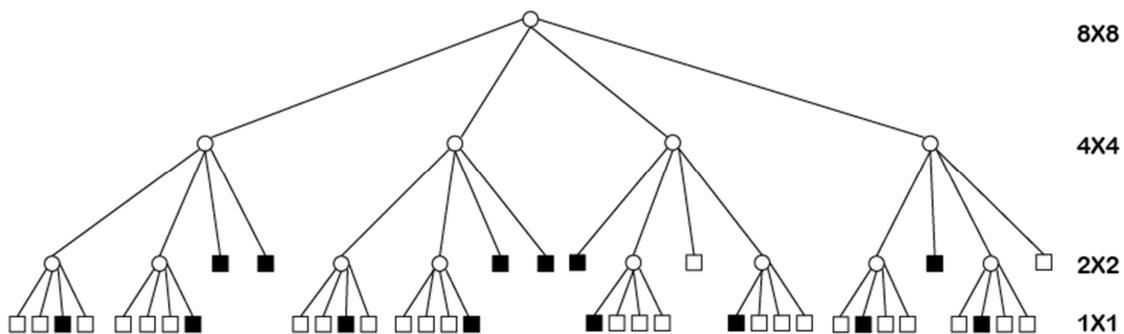


Berang-berang si pembajak mempunyai peta yang sangat besar sehingga harus dipotong-potong dalam potongan kecil. Setiap potongan peta berukuran 8 x 8 petak seperti gambar. Malangnya, kapal si pembajak terlalu kecil sehingga tak dapat membawa semua potongan sekaligus. Untungnya, si pembajak sangat cerdas untuk mendokumentasikan setiap potongan dalam catatannya.

1. Jika semua petak dalam potongan peta sama warnanya, dia mencatat sebagai persegi dengan warna petak tersebut
2. Atau jika tidak, ia menandai dengan lingkaran dan membagi potongan peta menjadi 4 bagian yang sama seperti pada Gambar 2.
3. Ulangi proses sampai semua petak ditandai (Gambar 4).



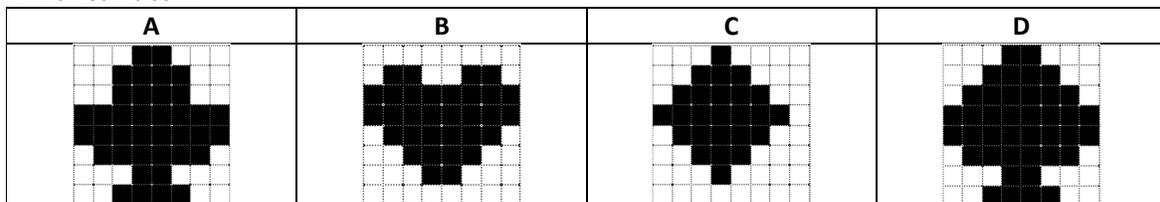
Berikut ini adalah catatan yang ada di buku bebras si pembajak.



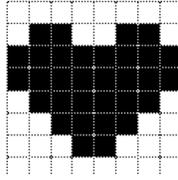
Tantangan:

Peta yang mana yang cocok dengan catatan tsb?

Pilihan Jawaban:

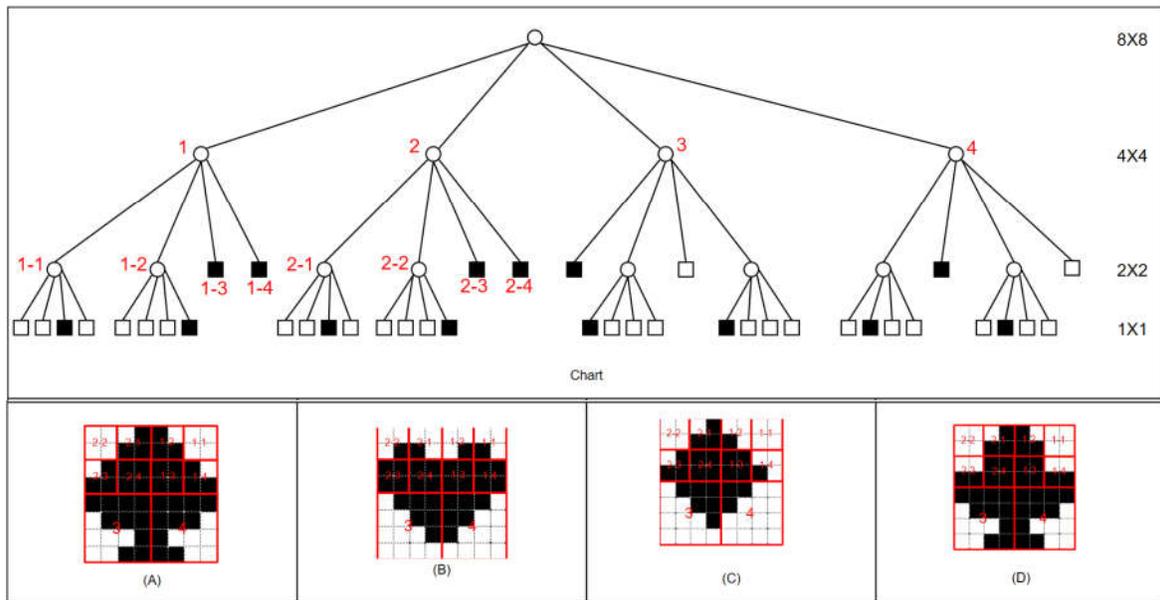


Jawaban:

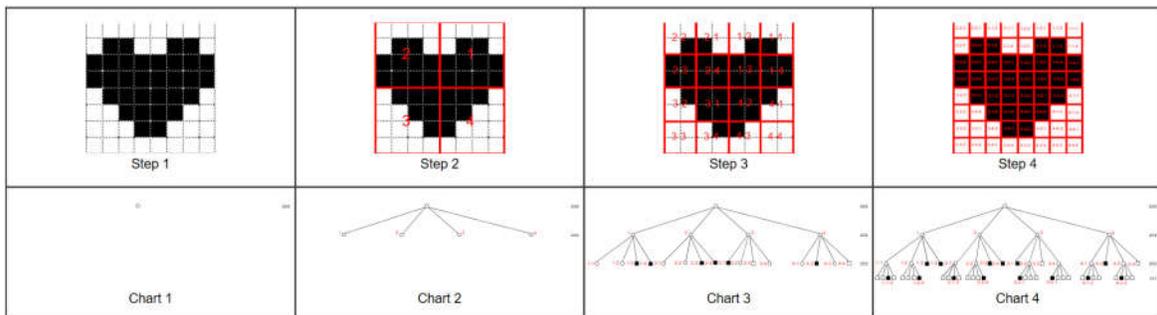


Jawaban yang paling tepat adalah B.

Dalam bagan, kita dapat menandai titik-titik yang mewakili daerah 4x4 dengan angka 1 hingga 4, dan menandai sub-titik dari titik 1 dengan angka 1-1 hingga 1-4, dan seterusnya. Posisi daerah yang sesuai di peta dan titik-titik dalam bagan ditunjukkan seperti di bawah ini. 1-3, 1-4, 2-3, dan 2-4 ditandai dengan warna hitam dalam bagan, yang berarti bahwa warna dari keempat wilayah 2x2 dalam peta berwarna hitam. Hanya (B) yang memenuhi kriteria ini.



Berikut ini adalah proses untuk merepresentasikan informasi potongan peta dalam bagan sesuai dengan langkah-langkah yang ditentukan dalam tugas. Ini memungkinkan Anda untuk mengetahui secara pasti mengapa peta (B) adalah jawaban yang benar. Dengan cara ini, ruang yang diperlukan untuk menyimpan informasi peta dapat dikurangi.



Ini Informatika!

Bagan 4-wilayah disebut "**Quadtree**" dalam Informatika. *Quadtree* adalah struktur data pohon di mana setiap simpul memiliki empat anak yang terhubung dengan simpul dari atas ke bawah. Semua bentuk *quadtree* memiliki beberapa fitur umum:

- Mereka menguraikan ruang menjadi sel-sel yang bisa beradaptasi.

- Setiap sel (atau *bucket*) memiliki kapasitas maksimum. Ketika kapasitas maksimum tercapai, bucket akan terbelah.
- Direktori *tree* mengikuti dekomposisi spasial dari *quadtree*.

Ada beberapa penggunaan *quadtrees* yang umum, seperti representasi gambar (contoh ini), pemrosesan dan kompresi gambar, pembuatan mesh, ... dll. Anda dapat menemukan informasi lebih lanjut di <https://en.wikipedia.org/wiki/Quadtree>

Authorship

2018-04-09 Yin-yao Kao (Taiwan), joseph.tnfsh@gmail.com: Task Proposal

2018-04-30 Yin-yao Kao (Taiwan), joseph.tnfsh@gmail.com: Task Revision

2018-05-08 Ungyeol Jung (KR), purnagi@gmail.com: Task Improvement

2018-05-10 Gabriela Stupuriene (LT), gabriele.stupuriene@mii.vu.lt: Task Improvement

2018-05-10 Seul-ki Kim (KR), tmfrlska85@gmail.com: New Graphics

License

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

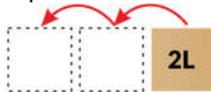
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0). Visit: <http://creativecommons.org/licenses/by-sa/4.0/>



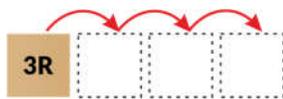
Berang-berang Yudi senang bermain lompat petak. Terdapat 8 petak yang diberi nomor dari 1 s.d. 8. Setiap petak berisi 1 kotak yang ditandai dengan salah satu dari tiga aturan melompat.

Contoh:

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

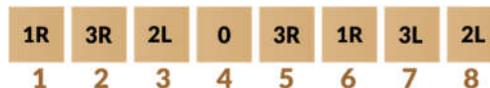


2. Gerakan ke kanan: Misalnya sebuah kotak ditandai dengan "3R" berarti ia harus melompat ke kanan sebanyak 3 petak, lalu menandai petak akhir lompatannya:



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

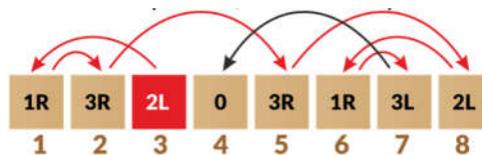
Pilihan Jawaban:

- A. 2
- B. 3
- C. 5
- D. Tidak mungkin mengunjungi semua petak.

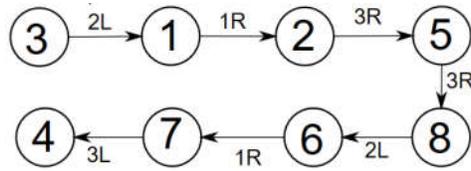
Jawaban:

Jawaban yang benar adalah 3.

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



Kita juga bisa menggambar ini sebagai graf, dengan label simpul menjadi petak, dan label jalur penghubung adalah "cara bergerak di antara kolom". Graf ini dapat ditarik mulai dari simpul mana saja, dan selesai ketika semua simpul 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 graf terarah, kita mencari simpul "head" atau "parent" dari koleksi ini.

Mengikuti serangkaian petunjuk penting dalam manajemen memori oleh sistem operasi (atau pengumpulan sampah 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 permainan 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

2018-04-09 Troy Vasiga (Canada), troy.vasiga@uwaterloo.ca:

New graphics provided by Eslam Wageed (Egypt), eslamwageed@gmail.com: boxes.png, boxes.swf,

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, henryasmo@gmail.com

Lisence

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

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

Visit: <https://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:

Jawaban yang benar adalah: 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

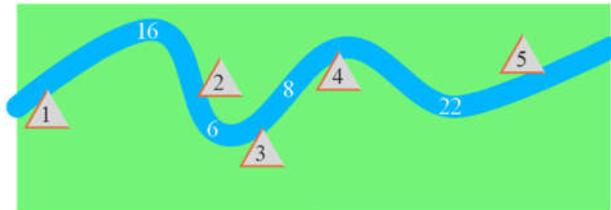
(2018-05-10), graphics Darija Dasović Rakijašić, darija.dasovic-rakijasic@skole.hr

Lisence

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

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

Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



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 yang paling minimum

Tantangan:

Pilih dua lokasi mana yang harus dipilih untuk membangun rumah?

Pilihan Jawaban:

- A. 1
- B. 2
- C. 3
- D. 4
- E. 5

Jawaban:

Jawaban yang tepat adalah 5 dan 2.

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

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

Waktu maksimum akan menjadi minimal dengan memilih gudang makanan 5 dan 2: waktu penyelamatan untuk gudang 1 adalah 16 menit; waktu untuk gudang 3 adalah 6, dan waktu penyelamatan untuk gudang 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 (rumah) dapat dibuka (di sini tanpa biaya tetap!). Dalam persoalan ini juga, kita ingin meminimalkan bukan jumlah waktu, tetapi waktu maksimum untuk sampai ke gudang 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

Lisence

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

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

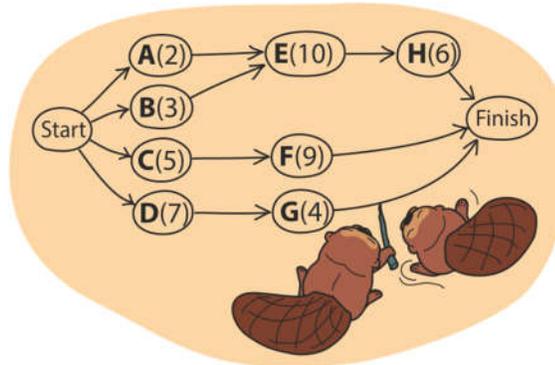
Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



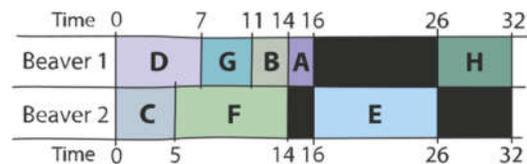
Dua Berang-berang Pekerja

PENEGAK (SMA)
I-2018-LT-04-X-011-5

Dua berang-berang sedang membangun sebuah bendungan sehingga perlu mengerjakan 8 tugas (memotong pohon, memotong cabang, mengalirkan kayu, merakit batang, dll. Setiap tugas dinomori dari A s.d. H: A(2), B(3), C(5), D(7), E(10), F(9), G(4), H(6). Angka dalam kurung menunjukkan berapa jam dibutuhkan untuk menyelesaikan setiap tugas. Para bebras bekerja secara bersamaan. Dan urutan penyelesaian pekerjaan diberikan dalam skema berikut:



Jika ada lebih dari satu tugas yang harus dikerjakan, maka yang dipilih adalah yang terbesar. Misalnya, para bebras dapat mengerjakan dengan urutan seperti tabel berikut:



Jika urutannya demikian, maka pekerjaan akan selesai dalam 32 jam. Tapi, masih ada cara lain untuk menyelesaikan pekerjaan tsb.

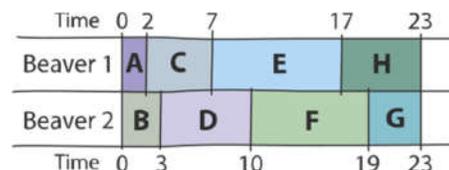
Tantangan: tentukan waktu yang paling singkat untuk membangun bendungan ini. Isikan sebuah bilangan bernilai antara 1 s.d. 99.

Jawaban:

Jawaban yang tepat adalah 23.

Gambar dalam pertanyaan menunjukkan jadwal dari dua berang-berang. Kita dapat melihat bahwa berang-berang pertama tidak bekerja untuk waktu yang relatif lama (8 jam), dan berang-berang kedua menganggur selama 6 jam. Akan lebih baik jika mereka bekerja sepanjang waktu.

Strategi yang akan kita gunakan di sini adalah untuk memastikan bahwa dua tugas terbesar E (10) dan F (9) tidak dilakukan oleh berang-berang yang sama. Berikut adalah gambar dari satu jadwal tertentu yang memungkinkan itu.



Kita akan melihat bahwa bendungan dapat dibangun dalam 23 jam!
Penting juga untuk mengetahui bahwa jadwal khusus ini memungkinkan pekerjaan dilakukan sesingkat mungkin karena berangkat-berang tidak pernah menganggur.

Ini Informatika!

Untuk beberapa contoh masalah, strategi berangkat-berang ("pilih sisa terbesar") akan menghasilkan waktu tersingkat. Untuk contoh masalah lain (seperti ini) strategi memilah tugas-tugas terbesar sepertinya akan berhasil. Namun, untuk masing-masing strategi ini kita dapat menemukan contoh masalah yang tidak bekerja dengan baik. Satu-satunya cara yang terjamin untuk menemukan jadwal yang menghasilkan waktu tersingkat adalah dengan mencoba semua yang solusi yang mungkin, kecuali ada batasan khusus pada masalahnya.

Ini tidak praktis dalam situasi dunia nyata; mungkin diperlukan lebih banyak sumber daya komputer untuk menemukan jadwal sempurna daripada satu berangkat-berang untuk membangun seluruh bendungan sendiri!

Dalam tugas ini, sebuah contoh masalah dipilih untuk menentukan strategi mana ("pilih sisa terbesar") yang tidak berfungsi. Dalam hal ini kita harus mempertimbangkan keseluruhan contoh masalahnya, bukan secara membabi buta mengikuti strategi sederhana. Namun, untuk kebanyakan masalah, strategi "serakah" berangkat-berang bisa jadi cukup baik, dan memiliki keuntungan bahwa jadwal dibuat dengan sangat cepat sehingga berangkat-berang langsung bekerja.

Menemukan contoh masalah untuk menyebabkan strategi berkinerja buruk (seperti yang dilakukan dalam persiapan tugas ini) adalah seni nyata dalam informatika, yang mengharuskan seseorang untuk memahami strategi. Ini adalah keterampilan yang diperlukan jika seseorang ingin tahu waktu menjalankan program komputer yang justru terburuk, yang juga dikenal sebagai analisis algoritma dan digunakan bidang komputasi teori kompleksitas.

[https://en.wikipedia.org/wiki/Scheduling_\(computing\)](https://en.wikipedia.org/wiki/Scheduling_(computing))

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

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

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

Authorship

2018-04-09 Valentina Dagiene, Lithuania: valentina.dagiene@mii.vu.lt

Lisence

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

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

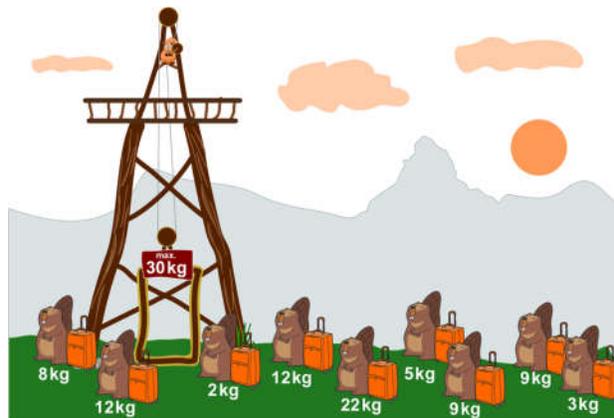
Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Lift Pengangkut Barang

PENEGAK (SMA)
I-2018-CH-07b

Sekumpulan barang-barang perlu membawa barang menggunakan sebuah lift pengangkut barang ke atas. Hari sudah malam, dan layanan lift akan dihentikan. Petugas hanya memberi kesempatan untuk dua kali naik. Kapasitas angkut lift untuk sekali jalan adalah 30 kg.



Tantangan:

Aturlah sehingga sebanyak mungkin barang yang bisa diangkut dengan hanya dua kali naik? Pilih berat barang yang akan diangkut di lift pertama dan kedua.



Jawaban:

Jawaban yang tepat adalah:

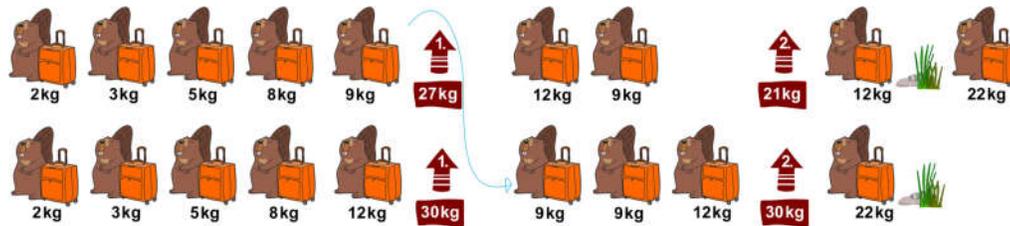
Lift Pertama: 9kg, 9kg, 12kg.

Lift Ke dua: 2kg, 3kg, 5kg, 8kg, 12kg.

Penting untuk pemrograman: Distribusi kedua grup barang-barang harus dimungkinkan di lift 1 dan 2, atau 2 dan 1. Urutan barang-barang di dalam lift harus tidak relevan.



Ide pertama yang terlintas dalam pikiran biasanya adalah memasukkan sebanyak mungkin barang yang paling ringan ke lift pertama: $2 + 3 + 5 + 8 + 9 = 27$ kg dan ke dalam kabin kedua $9 + 12 = 21$ kg (7 barang-barang). Tetapi sebenarnya kita bisa mengangkut 8 barang-barang berikut ke dalam lift:



Caranya: tukarkan barang-berang 9 kg dengan barang-berang 12 kg di lift pertama (total berat lift pertama 30 kg), sehingga kita dapat menggunakan barang-berang 9 kg untuk kabin kedua (total berat kabin kedua 30 kg).

Ini Informatika!

Masalah ini memiliki terlalu banyak kemungkinan, dan tidak mungkin kita bisa memeriksa semuanya dalam waktu yang singkat.

Kita harus menemukan solusi 'terbaik' untuk masalah ini, walaupun ini mungkin tidak selalu optimal. Dalam Ilmu Komputer kita sebut ini adalah masalah yang tidak dapat diselesaikan secara praktis. Namun kita bisa mengikuti strategi yang praktis, dimulai dengan mencoba menempatkan barang-berang sebanyak mungkin ke dalam lift pertama. Dalam informatika rencana tersebut mengarah ke solusi yang baik tetapi mungkin bukan yang terbaik yang disebut heuristik.

Authorship

Urs Hauser, urs.hauser@inf.ethz.ch, Switzerland
 Juraj Hromkovic, juraj.hromkovic@inf.ethz.ch, Switzerland
 Regula Lacher, regula.lacher@inf.ethz.ch, Switzerland
 Jacqueline Staub, jacqueline.staub@inf.ethz.ch, Switzerland
 Vaidotas Kinčius, info@vectorsketch.eu, Lithuania
 Martin Guggisberg, martin.guggisberg@fhnw.ch, Switzerland
 Andrea Maria Schmid, andrea.schmid3@phlu.ch, Switzerland
 Doris Reck, doris.reck@vsluzern.ch, Switzerland
 Susanne Datzko, susanne@datzko.ch, Switzerland
 Christian Datzko, christian@datzko.ch, Switzerland

Lisence

Copyright © 2018 Bebras – International Challenge on Informatics and Computational Thinking.
 This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.
 Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Sebuah alat untuk melakukan diagnosa harus menggoncang spesimen secara berulang-ulang. Alat ini bekerja berdasarkan sebuah program komputer, yang ditulis dalam beberapa baris yang diberi nomor. Alat membaca program baris demi baris, dan mengeksekusinya segera setelah membaca.

Jika baris mengandung perintah **go to X**, maka alat akan langsung ke baris X dan meneruskan membaca serta mengeksekusinya

Program mampu untuk:

- menyimpan sebuah nilai bilangan dalam lokasi A dengan instruksi “**set**”,
- menambahkan 1 pada nilai yang disimpan pada lokasi A dengan instruksi “**add**”,
- dan membandingkan nilai A dengan sebuah bilangan lain (**=**, **<**, **≤**, **>**, **≥**, **≠**).

Tantangan: Berapa kali alat akan menggoncang spesimen jika prosedurnya ditulis dengan program sebagai berikut:

1. set A to 0
2. add 1 to A
3. go to 6
4. jika A = 60 go to 8
5. set A to 0
6. add 1 to A
7. go to 2
8. ulangi A kali menggoncang spesimen
9. stop

Pilihan Jawaban:

- A. Spesimen digoncang dua kali.
- B. Spesimen digoncang satu kali.
- C. Spesimen digoncang 60 kali.
- D. Prosedur tidak akan pernah berhenti dan tidak pernah mengguncang spesimen.

Jawaban:

Jawaban yang benar adalah D. Prosedur tidak akan pernah berhenti dan tidak pernah mengguncang spesimen.

Program ini selalu memerintah untuk melompat dari baris 3 ke 6 dan dari baris 7 ke 2. Kecuali pada awalnya, program hanya mengunjungi baris No. 2, 3, 6, 7. Instruksi untuk menggoyang spesimen ada di baris No. 8, yang tidak pernah dikunjungi. Ini berarti perangkat tidak akan pernah mengguncang apa pun sesuai dengan program, jadi jawaban yang benar adalah d). Selain itu, instruksi pada saluran No. 9 tidak pernah dieksekusi, sehingga program berlanjut selamanya.

Ini Informatika!

Bahasa pemrograman pertama, yang dikembangkan pada tahun '40-an dan '50 -an, seperti yang nampak di tugas ini disebut bahasa rakitan. Baris program diberi nomor dan perintah khusus, yang dikenal sebagai instruksi, digunakan untuk melompat ke baris yang berbeda dari yang berikutnya. Sangat sulit untuk membaca program-program ini dan menemukan kesalahan tetapi mudah membuatnya. Bahasa pemrograman seperti itu rawan kesalahan sehingga bahasa pemrograman yang lebih modern dikembangkan mulai tahun 50-an. Bahasa-bahasa modern ini tidak berorientasi pada garis, melainkan mengandung struktur seperti loop, prosedur dan pilihan.

<https://homepages.cwi.nl/~storm/teaching/reader/Dijkstra68.pdf>

Authorship

2018-02-01 Jiří Vaníček (Czechia), vanicek@pf.jcu.cz Task Proposal
Judith Lin, dith007@gmail.com, Taiwan.
Christian Datzko, christian.datzko@informatik-biber.ch, Switzerland

Lisence

Copyright © 2018 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.
Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Pada suatu hari yang cerah, Maya, David, Iva, dan Marko bermain sepak bola. Malangnya, salah satu melempar bola dan memecahkan kaca kelas. Bu Guru ingin tahu siapa yang menyebabkan kaca jendela tsb pecah. Bu Guru mengenal dengan baik bahwa tiga di antara anak tersebut tidak pernah bohong. Tapi ia tidak yakin siapa yang bersalah.



Anak-anak tersebut berkata secara berurutan :

- Marko: Bukan saya yang memecahkan kaca
- Iva: Marko atau David yang memecahkan kaca
- Maya: David yang memecahkan kaca
- David: bukan saya, Maya bohong!

Tantangan:

Siapa yang memecahkan kaca jendela?

Pilihan Jawaban:

- A. David
- B. Marko
- C. Maya
- D. Iva

Jawaban:

Jawaban yang benar adalah David.

Hal pertama yang kita temukan adalah bahwa pernyataan dari Maya dan David tidak bisa keduanya benar atau keduanya berbohong. Karena itu, salah satu dari mereka mengatakan yang sebenarnya, dan yang satunya berbohong.

Ada dua cara yang berbeda, sama benarnya, yang bisa kita gunakan. Penting untuk mengetahui bahwa kedua pendekatan ini bisa dipakai:

----- Pendekatan 1 -----

- (a) Jika Maya mengatakan yang sebenarnya, maka hanya David yang berbohong.
- (b) Jika David mengatakan yang sebenarnya, maka Maya dan salah satu dari Iva atau Marko berbohong, tetapi hanya ada satu pembohong di kelompok.

Ada dua kemungkinan (a) dan (b) bahwa David yang memecahkan jendela.

----- Pendekatan 2 -----

Atau, secara lebih umum, kita dapat menyelesaikan masalah dengan cara berikut:

- (a) Jika Maya berbohong ketika mengatakan bahwa "David memecahkan jendela", maka itu berarti yang lain harus mengatakan yang sebenarnya. (Kita tahu itu karena guru Ana mengenal murid-muridnya dan dia tahu itu mereka bertiga selalu mengatakan yang sebenarnya.) Dalam hal itu, Marko mengatakan yang sebenarnya ketika dia mengatakan bahwa dia tidak memecahkan jendela, dan sesuai dengan pernyataan Iva berarti bahwa David yang memecahkan jendela. Tapi, itu

bertentangan dengan pernyataan David, yang artinya bahwa ini bukan jawaban yang benar.
(b) Jika David berbohong, maka itu berarti yang lain harus mengatakan yang sebenarnya. Kalau begitu, Marko tidak memecahkan jendela. Iva menyatakan bahwa David memecahkan jendela, dan Maya mengatakan hal yang sama, jadi ini bisa jadi adalah jawaban yang benar.
Ada dua kemungkinan (a) dan (b) bahwa David memecahkan jendela.

Ini Informatika!

Dasar teori aljabar logis, yang mendasari semua pemrograman komputer, diciptakan pada 1854 oleh George Boole (1815-1864). Elemen dasar aljabar logis adalah pernyataan logis. Setiap pernyataan dapat dengan tegas ditentukan apakah itu benar atau salah.

Pernyataan benar: benar, T atau 1

Pernyataan salah: salah, F atau 0

Pernyataan-pernyataan tersebut dapat digabungkan secara bersama dalam istilah logis di mana pernyataan disebut operandi dan jenis operasi yang dilakukan antara operandi memberitahu kita operatornya. Operasi logika dasar terdiri dari satu atau dua operandi dan satu operator sedangkan operasi logis kompleks terdiri dari logika dasar atau dasar operasi. Tabel status (tabel kebenaran) menyatakan hubungan antar operandi yang tergantung pada logika operasi. Masing-masing, berdasarkan keaslian semua pernyataan yang termasuk dalam operasi, menentukan keasliannya.

Karena komputer terbuat dari sirkuit elektronik yang hanya membedakan dua kondisi stabil, Prinsip Aljabar Boolean (operasi, operand dan aturan hubungan logis) dapat diterapkan pada konstruksi dan analisis pekerjaan komputer digital.

Authorship

A lie has no legs 2018-HR-05-eng.odt, Last saved 2018-05-10 at 09:37:16 by
Sanja Pavlovic Šijanović, Croatia, sanja.pavlovic-sijanovic@skole.hr

Lisence

Copyright © 2018 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.
Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Twist and Turn

Tom adalah seorang berumur 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 kiri
	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 D4 D5 D6 E6 F6
- b) A1 B1 B2 B3 C3 D3 E3 E4 F4 F5 F6
- c) A1 B1 B2 B3 C3 D3 E3 F3 F4 F5 F6
- d) A1 B1 B2 B3 B4 C4 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 array dua dimensi yang digunakan untuk menggambarkan peta dan array satu dimensi yang digunakan untuk menggambarkan arah bergerak. Array berisi data yang tidak sama, masing-masing elemen adalah struktur data yang terdiri dari dua komponen: karakter dan angka. Untuk menyelesaikan tugas ini, anda harus membaca array dan melacak elemen matriks yang indeksnya ditemukan sebagai nilai dalam array. Meskipun masalahnya nampak sederhana, pembacaan elemen array dan elemen matriks secara simultan adalah sulit karena kondisi yang diberlakukan oleh nilai elemen matriks.

Anda dapat menemukan solusi dengan menerapkan pencarian pertama yang mendalam (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

Lisence

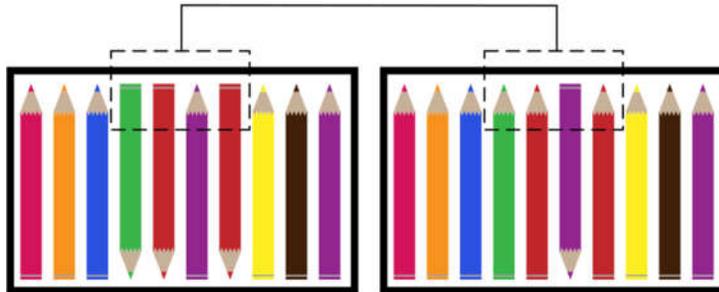
Copyright © 2018 Bebras – International Challenge on Informatics and Computational Thinking.

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

Visit: <https://creativecommons.org/licenses/by-sa/4.0/>

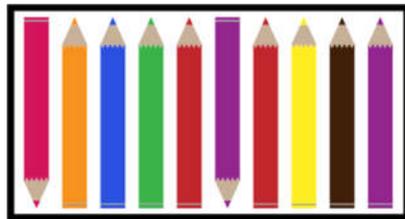


Ade mempunyai sekotak krayon. Beberapa krayon ada yang menghadap ke atas, ada yang menghadap ke bawah. Menurut Ade, sekotak krayon akan “indah dipandang” kalau semua krayon di dalamnya menghadap ke arah yang sama. Pada satu langkah, ia dapat membalikkan sederet krayon dalam satu baris; setelah melakukan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

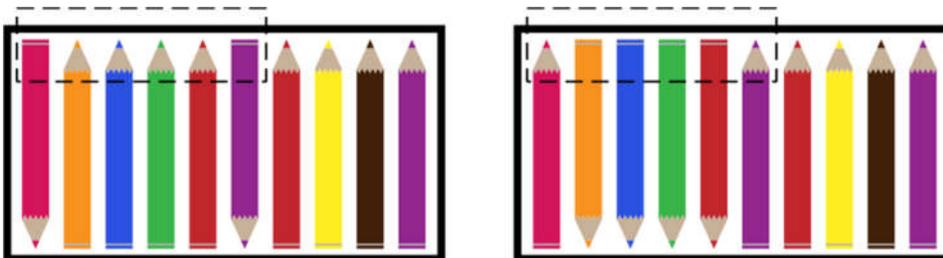


Jawaban:

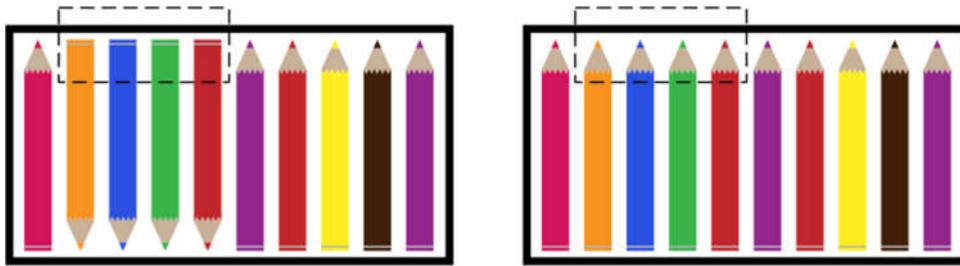
Jawaban yang tepat adalah 2.

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 pemrogram 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.

Authorship

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

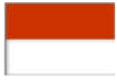
Task Proposal based on idea: <https://www.codechef.com/problems/ADACRA>

Lisence

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

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

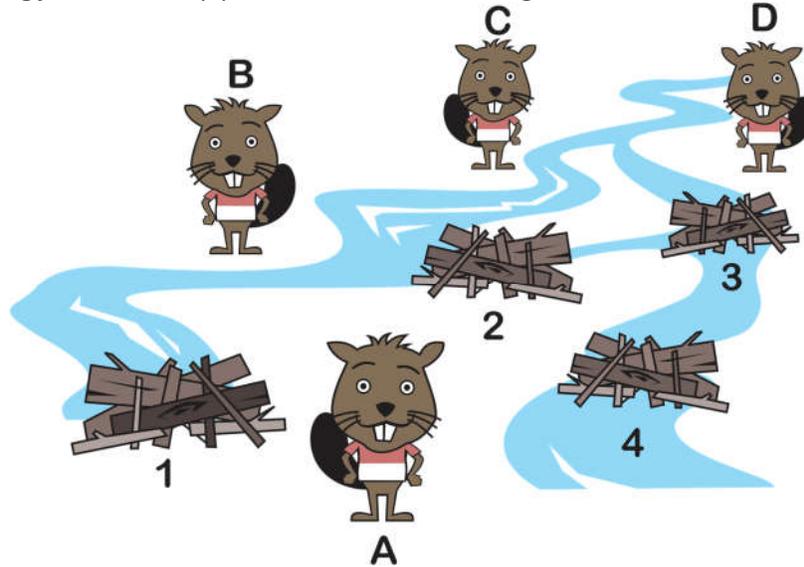
Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Bendungan 2

PENEGAK (SMA)
2018-ID-01a

Bapak Walikota Bebras harus memelihara 4 (empat) bendungan yang tersebar di kota setiap hari. Untuk pemeliharaan tersebut, 4 (empat) bebras A, B, C dan D 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 ditentukan oleh jarak yang ditempuh oleh bebras ke bendungan. Agar paling murah, Pak walikota ingin agar total jarak yang harus ditempuh oleh keempat bebras tersebut minimal. Jarak dari rumah setiap bebras ke setiap bendungan (dalam meter) diberikan pada tabel sebagai berikut:

	Bendungan 1	Bendungan 2	Bendungan 3	Bendungan 4
A	185	145	143	190
B	130	125	175	225
C	50	50	100	75
D	220	186	185	225

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

Jawaban:

Jawaban yang benar adalah 534.

Ini Informatika!

Ini masalah kombinatorik. Adaptasi dari *Bipartit Matching*. Dengan cara naif seperti untuk matriks M ukuran 3x3, diperoleh 24 kemungkinan sbb.

$$M[A,1]+M[B,2]+M[C,3]+M[D,4]=185+125+100+225=635$$

$$M[A,1]+M[B,2]+M[C,4]+M[D,3]=185+125+75+185=570$$

$$M[A,1]+M[B,3]+M[C,2]+M[D,4]=185+175+50+225=635$$

$$M[A,1]+M[B,3]+M[C,4]+M[D,2]=185+175+75+186=621$$

$$M[A,1]+M[B,4]+M[C,2]+M[D,3]=185+225+50+185=645$$

$$M[A,1]+M[B,4]+M[C,3]+M[D,2]=185+225+100+186=696$$

$$M[A,2]+M[B,1]+M[C,3]+M[D,4]=145+130+100+225=600$$

$M[A,2]+M[B,1]+M[C,4]+M[D,3]=145+130+75+185=535$
 $M[A,2]+M[B,3]+M[C,1]+M[D,4]=145+175+50+225=595$
 $M[A,2]+M[B,3]+M[C,4]+M[D,1]=145+175+75+220=615$
 $M[A,2]+M[B,4]+M[C,1]+M[D,3]=145+225+50+185=605$
 $M[A,2]+M[B,4]+M[C,3]+M[D,1]=145+225+100+220=690$
 $M[A,3]+M[B,1]+M[C,2]+M[D,4]=143+130+50+225=548$
 $M[A,3]+M[B,1]+M[C,4]+M[D,2]=143+130+75+186=534$
 $M[A,3]+M[B,2]+M[C,1]+M[D,4]=143+125+50+225=543$
 $M[A,3]+M[B,2]+M[C,4]+M[D,1]=143+125+75+220=563$
 $M[A,3]+M[B,4]+M[C,1]+M[D,2]=143+225+50+186=604$
 $M[A,3]+M[B,4]+M[C,2]+M[D,1]=143+225+50+220=638$
 $M[A,4]+M[B,1]+M[C,2]+M[D,3]=190+130+50+185=555$
 $M[A,4]+M[B,1]+M[C,3]+M[D,2]=190+130+100+186=606$
 $M[A,4]+M[B,2]+M[C,1]+M[D,3]=190+125+50+185=550$
 $M[A,4]+M[B,2]+M[C,3]+M[D,1]=190+125+100+220=635$
 $M[A,4]+M[B,3]+M[C,1]+M[D,2]=190+175+50+186=601$
 $M[A,4]+M[B,3]+M[C,2]+M[D,1]=190+175+50+220=635$
 Minimum dari semua 534

Untuk ukuran lebih besar terdapat sejumlah penghitungan berulang yang sama. Diperlukan strategi fungsi rekursif (DP) sbb. Misalnya untuk kasus 4x4 tsb.

Untuk 2 hal (A,B) dipetakan ke 2 item dari 4 kemungkinan item (1,2,3,4):

$f(\{A,B\},\{1,2\}) = \min(M[A,1]+M[B,2], M[A,2]+M[B,1]) = \min(185+125, 145+130) = 275$
 $f(\{A,B\},\{1,3\}) = \min(M[A,1]+M[B,3], M[A,3]+M[B,1]) = \min(185+175, 143+130) = 273$
 $f(\{A,B\},\{1,4\}) = \min(M[A,1]+M[B,4], M[A,4]+M[B,1]) = \min(185+225, 190+130) = 310$
 $f(\{A,B\},\{2,3\}) = \min(M[A,2]+M[B,3], M[A,3]+M[B,2]) = \min(145+175, 143+125) = 268$
 $f(\{A,B\},\{2,4\}) = \min(M[A,2]+M[B,4], M[A,4]+M[B,2]) = \min(145+225, 190+175) = 315$
 $f(\{A,B\},\{3,4\}) = \min(M[A,3]+M[B,4], M[A,4]+M[B,3]) = \min(143+225, 190+175) = 365$

Untuk 3 hal (A,B,C) dipetakan ke 3 item dari 4 kemungkinan (1,2,3,4):

$f(\{A,B,C\},\{1,2,3\}) = \min(f(\{A,B\},\{1,2\})+M[C,3], f(\{A,B\},\{1,3\})+M[C,2], f(\{A,B\},\{2,3\})+M[C,1]) = \min(275+100, 273+50, 268+50) = 318$
 $f(\{A,B,C\},\{1,2,4\}) = \min(f(\{A,B\},\{1,2\})+M[C,4], f(\{A,B\},\{1,4\})+M[C,2], f(\{A,B\},\{2,4\})+M[C,1]) = \min(275+75, 310+50, 315+50) = 350$
 $f(\{A,B,C\},\{1,3,4\}) = \min(f(\{A,B\},\{1,3\})+M[C,4], f(\{A,B\},\{1,4\})+M[C,3], f(\{A,B\},\{3,4\})+M[C,1]) = \min(273+75, 310+100, 365+50) = 348$
 $f(\{A,B,C\},\{2,3,4\}) = \min(f(\{A,B\},\{2,3\})+M[C,4], f(\{A,B\},\{2,4\})+M[C,3], f(\{A,B\},\{3,4\})+M[C,2]) = \min(268+75, 315+100, 365+50) = 343$

Untuk 4 hal dipetakan ke 4 item:

$f(\{A,B,C,D\},\{1,2,3,4\}) = \min(f(\{A,B,C\},\{1,2,3\})+M[D,4], f(\{A,B,C\},\{1,2,4\})+M[D,3], f(\{A,B,C\},\{1,3,4\})+M[D,2], f(\{A,B,C\},\{2,3,4\})+M[D,1]) = \min(318+220, 350+185, 348+186, 343+220) = \min(538, 535, 534, 563) = 534$

Authorship

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

Graphics by Timotius Kevin Levandi, kevin.levandi@gmail.com

Lisence

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

Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Tugas Satu Jam

PENEGAK (SMA)
I-2018-ID-02ab

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

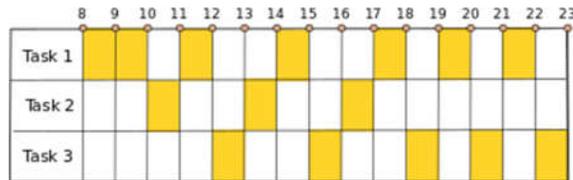
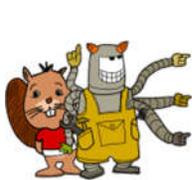
Dalam satu jam, si robot hanya dapat mengerjakan satu tugas. Pada akhir setiap jam, dia mengecek apakah ada sebuah tugas baru:

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

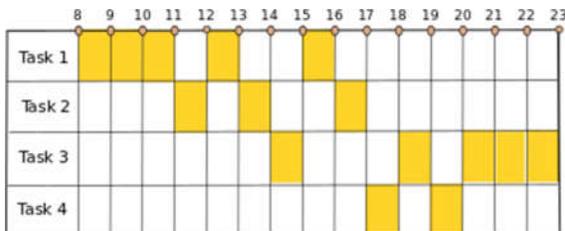
Kapan Tugas-4 akan selesai?

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 dan menggambarkan 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

Vaidotas Kinčius, info@vectorsketch.eu (Licence to Bebras according CC, see on the bottom)

Lisence

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

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

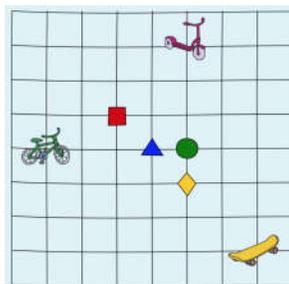
Visit: <https://creativecommons.org/licenses/by-sa/4.0/>



Tiga sekawan Berang-berang

PENEGAK (SMA)
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 yang benar:

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: $3+4+5 = 12$

 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 soal ini, tambahkan semua jarak yang dicakup oleh tiga karakter untuk masing-masing berangkat-berangkat.

Jarak yang terpendek adalah jawaban yang benar. Menggunakan metode ini, Anda menggunakan pencarian lokal 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/>