

SILABUS 2018

OLIMPIADE KOMPUTER INTERNASIONAL UNTUK SELEKSI OLIMPIADE SAINS NASIONAL TINGKAT KABUPATEN/KOTA, PROVINSI, DAN NASIONAL



**INFORMATIKA/
KOMPUTER**



Kementerian Pendidikan dan Kebudayaan
Direktorat Jenderal Pendidikan Dasar dan Menengah
Direktorat Pembinaan Sekolah Menengah Atas

SILABUS DAN KISI-KISI SELEKSI OLIMPIADE SAINS NASIONAL BIDANG INFORMATIKA/KOMPUTER

Versi Januari 2018

A. Pengantar

A.1. Olimpiade Sains Nasional

Pada saat ini Kementerian Pendidikan dan Kebudayaan menyelenggarakan Olimpiade Sains Nasional (OSN) tingkat SMA yang terdiri dari 9 bidang olimpiade sains. Satu di antaranya adalah bidang Informatika/Komputer. Pemilihan peserta yang akan bertanding di OSN dilakukan melalui seleksi berjenjang dan serentak di seluruh Indonesia, yaitu:

- tingkat kabupaten/kota (Olimpiade Sains tingkat Kabupaten/Kota – OSK), kemudian
- tingkat provinsi (Olimpiade Sains tingkat Provinsi – OSP).

Seleksi di tingkat provinsi (OSP), pada umumnya diikuti oleh sekitar 1500 siswa peserta seleksi pertahunnya untuk masing-masing bidang olimpiade. Sedangkan di tingkat kabupaten/kota tentunya sekian kali lebih banyak lagi (estimasi kasar ada di atas 8-12 ribuan siswa¹). Hasil dari seleksi tingkat provinsi menentukan siapa yang akan menjadi salah seorang dari ke 70 hingga 90 siswa peserta OSN.

Selain sebagai ajang prestasi tingkat nasional, OSN bertujuan juga untuk mendapatkan calon peserta pembinaan dan seleksi lebih lanjut hingga dipilih empat siswa terbaik untuk menjadi anggota TOKI (Tim Olimpiade Komputer Indonesia). Mereka itulah yang akan mewakili negara dan bangsa untuk bertanding di tingkat dunia yaitu International Olympiad in Informatics (IOI).

A.2. International Olympiad in Informatics

IOI adalah ajang kompetisi pemrograman di tingkat Internasional yang sudah berlangsung sejak 1985. Indonesia mulai mengikuti IOI sejak 1995. Saat ini IOI diikuti oleh lebih dari 80 negara (termasuk semua negara maju) sehingga IOI merupakan lomba paling akbar dalam bidang ini untuk tingkat SMA.

Pada awalnya IOI sendiri adalah lomba murni pemrograman semata berdasarkan masalah-masalah yang sederhana. Para peserta dari berbagai negara secara perseorangan berusaha menyelesaikan sejumlah masalah dalam waktu yang singkat dengan membuat program penyelesaian masalah. Program yang dihasilkan diuji dengan sejumlah data test (test case) yang mewakili sejumlah kondisi yang mungkin

¹ Ini hanya perkiraan kasar saja karena di tingkat kabupaten/kota, penyelenggaraan beserta proses seleksi diserahkan ke masing-masing kabupaten/kota yang bersangkutan sehingga data peserta tidak tercatat dengan lengkap. Sementara, di tingkat provinsi, proses seleksi dilakukan di pusat sehingga bisa diketahui jumlah keseluruhan peserta.

dari input soal tersebut. Program yang dibuat peserta dinilai dari berapa banyak test case yang berhasil dijawab dengan benar oleh program tersebut. Nilai akhir peserta adalah jumlah nilai yang diperoleh dari setiap program yang dibuatnya. Peringkat peserta diurutkan berdasar nilai tersebut dan 1/12 (atau 8.33%) dari semua peserta pada peringkat teratas peserta mendapatkan medali emas, 1/6 (atau 16.67%) berikutnya mendapatkan medali perak dan 1/3 (atau 33.33%) mendapatkan medali perunggu. Sisanya terdapat 50% yang tidak mendapatkan medali. Peringkat negara disusun atas total perolehan medali peserta.

Dalam perjalanannya problem-problem yang diberikan mengalami peningkatan tingkat kesulitannya terutama sejak akhir tahun 90-an, hingga pada saat ini pemrograman hanya satu aspek kecil semata di dalam lomba ini. Dengan demikian, aspek utama yang diuji adalah kemampuan menyelesaikan masalahnya sendiri. Sehingga bisa dikatakan bahwa kompetisi IOI adalah menguji kemampuan peserta dalam problem solving dengan pemrograman komputer². Setiap peserta dalam waktu yang amat terbatas harus mengerjakan sejumlah masalah yang diberikan dengan menyusun program yang menyelesaikan masalah tersebut.

B. Karakteristik Materi Uji

B.1. Tingkat IOI

Secara umum penyelesaian masalah di tingkat IOI memerlukan aspek-aspek dalam proses berfikir sebagai berikut.

1. Peserta harus mampu membaca deskripsi soal (termasuk input-proses-output) yang dinarasikan sebagai suatu cerita yang di dalamnya terkandung suatu permasalahan yang hendak diselesaikan.
2. Terkait dengan itu, diperlukan juga pemahaman logika yang baik. Agar berdasarkan deskripsi tersebut peserta mampu menyusun model/abstraksi permasalahan. Model dapat berupa interrelasi antar entitas sebagai suatu graf atau bahkan sudah menjadi lebih matang lagi sebagai suatu model atau fungsi rekurens
3. Menemukan metoda dalam penyusunan algoritma berdasarkan model/abstraksi sebelumnya
4. Mampu melakukan optimasi model penyelesaian masalah di aspek 2 dan 3 tersebut untuk mencapai efisiensi algoritma terbaik. Hanya sekedar solusi naif saja tidak dapat mencapai nilai maksimum)
5. Konversi rancangan algoritma di atas menjadi program serta evaluasi hasil kerja pemrograman di atas berdasarkan seluruh kemungkinan test case yang akan diberikan.
 - a. mendeduksi proses dari test case (Input - Output)
 - b. mengenali variabilitas test case (kasus ekstrim, kasus sederhana)
6. Melakukan manajemen waktu, memelihara ketelitian dan stamina dalam mengerjakan hal-hal di atas (tahan terhadap presure keterbatasan waktu dan

² Harap bagian yang digarisbawahi tersebut dipahami secara lengkap; bukan HANYA menguji kemampuan membuat program komputer, bukan pula HANYA menguji kemampuan menyelesaikan masalah, tetapi KEDUANYA!!!

memiliki endurance, keuletan dan ketelitian untuk tidak meloloskan sedikitpun kesalahan)

Sebagai catatan, kemampuan dalam penyusunan program hanyalah salah satu aspek saja, yang lebih sulit adalah dalam kelima aspek-aspek. Efisiensi akan ditentukan dari metodologi apa yang digunakan pada tahap ke 3.

B.2. Tingkat OSK/OSP

Proses seleksi idealnya adalah mengacu model IOI di atas yaitu problem solving dengan pemrograman. Namun, berbeda dengan bidang OSN lain seperti Fisika, Matematika, Kimia dan Biologi, bidang Informatika khususnya pemrograman belum menjadi pelajaran resmi. Kalaupun ada, hanya di sekolah-sekolah tertentu saja dan itupun belum tentu mengajarkan pemrograman. Dalam kurikulum TIK Diknas, pemrograman hanya diberikan dalam porsi yang amat kecil (sebagian besar adalah penggunaan perangkat lunak MS Office).

Oleh sebab itu, materi uji IOI “diterjemahkan” ke dalam materi yang menguji potensi akademis/skolastik tinggi yang relevan dengan aspek-aspek di atas. Diharapkan dari proses seleksi ini, siswa yang berpotensi walaupun belum mahir dalam pemrograman dapat terjaring untuk diberikan pembinaan yang intensif di Pelatnas.

Aspek yang sangat bergantung pada ketrampilan peserta dalam pemrograman dikurangi dan digantikan dengan materi uji “analisa dan logika” dan materi uji kemampuan algoritmika.

Tingkatan seleksi OSK-OSP dibedakan atas komposisi dari ketiga komponen materi uji:

- Kemampuan analitika/logika/aritmatika (nonprogramming)
- Kemampuan algoritmika (programming)

Komponen uji pemrograman tidak mungkin untuk diadakan sehingga digantikan dengan kemampuan dan algoritmika saja.³ Metoda pengujiannya pun tidak bisa dihindari bersifat test obyektif (pilihan ganda). Metoda ini memang banyak sekali kelemahannya yaitu memungkinkan jawaban asal tapi benar, namun, memungkinkan pemeriksaan yang segera dan efisien. Dampak negatif tersebut bisa dikurangi dengan pembuatan soal dan pilihan jawaban yang dirancang dengan matang. Komposisi analitika/logika di tingkat kabupaten/kota adalah yang paling besar.

Di tingkat provinsi pada dasarnya sama dengan di tingkat kabupaten/kota kecuali komposisi algoritmika diperbesar. Ini adalah untuk memacu peserta yang lolos di tingkat kabupaten/kota untuk memperdalam pemahamannya dan ketrampilan prakteknya dalam pemrograman.

B.3. Tingkat OSN

Materi uji semula adalah mirip dengan di tingkat IOI, namun terhubung belum siapnya sebagian besar peserta dalam hal praktek pemrograman maka materi uji pemrograman dikombinasikan dengan materi uji non programming seperti di OSK/OSP.

³ Uji pemrograman di tingkat provinsi, apalagi di tingkat kabupaten/kota, masih perlu beberapa tahun lagi hingga infrastruktur di setiap daerah sudah merata.

C. Kisi-kisi Materi Nonprogramming

C.1. Umum

Secara umum materi uji tertulis terbagi atas tiga komponen utama: materi uji analitika dan logika, materi uji aritmatika, dan materi uji algoritmika.

- Materi analitika yang bersifat logika bertujuan untuk menguji potensi akademis (skolastik) peserta namun sedapat mungkin memiliki relevansi yang tinggi dengan problem solving dan elemen penting dalam menguasai pemrograman komputer. Kemampuan ini merupakan faktor penting dalam memahami persoalan yang diberikan dan merancang algoritma penyelesaian masalahnya.
- Materi analitika yang bersifat aritmatika sebenarnya sejalan dengan analitika dan logika di atas, karena soal aritmatika disini bukan sekedar menguji ketrampilan dalam hitung-menghitung, tetapi lebih pada cara berpikir yang logis dan analitis namun dengan soal bertemakan aritmatika
- Materi algoritmika bertujuan untuk menguji kemampuan peserta dalam memahami dan menyusun suatu algoritma. Aspek-aspek yang terkait dengan pengetahuan dan bahasa pemrograman direduksi seminimal mungkin ke tingkat pseudocode.

Lebih rinci lagi ketiga kategori tersebut dijabarkan dalam sejumlah aspek sebagai berikut.

C.2. Tipe Soal untuk Menguji Deskripsi Soal

Soal berbentuk cerita untuk menguji kemampuan aspek pertama dan kedua dari proses berfikir IOI pada bagian B.1. (adaptasi dari soal IOI namun untuk dapat diselesaikan secara manual). Soal dibuat untuk mengukur:

- Kemampuan memahami dan mensimulasikan algoritma dalam cerita
- Kemampuan deduksi berdasarkan input menghasilkan output
- Kemampuan deduksi berdasarkan test case (input-output) menghasilkan pemahaman proses.
- Kemampuan menemukan kasus-kasus ekstrim
- Kemampuan optimasi
- Kemampuan menemukan model matematika dari soal

C.3. Tipe Soal Pemahaman Algoritma

Dalam menjawab soal-soal ini peserta harus bisa memahami algoritma yang diberikan dalam notasi pseudopascal dan menelusuri eksekusi algoritma. Kemampuan ini diperlukan sebagai indikator aspek ke-3 dari proses berpikir IOI pada bagian B.1. Jadi soal-soal dibuat untuk mengujur

- Kemampuan memahami konsep elemen konstruksi (if-then-else, loop dan variasinya)
- Kemampuan membaca algoritma secara menyeluruh
- Kemampuan mengeksekusi (termasuk rekursif) dan *process tracing* yang terjadi
- Kemampuan menkonstruksi (coding)

C. 4. Tipe soal Kemampuan Dasar Logika

Dalam kaitannya dengan bagian C.1. sejumlah pernyataan logika terkait dengan aspek-aspek sebagai berikut.

- implikasi
- 'jika dan hanya jika'
- kalkulus preposisi
- induksi-deduksi

Pertanyaan yang diajukan dikombinasikan dengan bagian C.1. untuk mengajukan pertanyaan-pertanyaan yang memerlukan penguasaan dasar logika tersebut.

C.5. Menguji kemampuan dasar Aritmatika

Bagaimanapun juga Aritmatika tidak dipisahkan dari problem solving. Namun supaya agak berbeda dari materi uji Olimpiade Matematika maka aspek aritmatika yang akan dipertanyakan adalah yang:

- berisikan unsur langkah-langkah komputasi
- menuntut kemampuan penyusunan model matematis
- keterkaitan dengan sifat dari deret bilangan
- menuntut kemampuan penyusunan model keterkaitan (graf)

C.6. Tipe Soal Kemampuan Dasar Penunjang

Dalam dunia komputasi (problem solving) elemen dasar (matematika) penunjangnya secara teoritis dikenal sebagai matematika diskret⁴. Pada umumnya soal-soal jenis ini akan mengajukan pertanyaan-pertanyaan

- Himpunan
- Aljabar logika
- Sifat Bilangan (deret)
- Finite State Machine
- Kombinatorik

Pemahaman yang diperlukan bukanlah pemahaman teoritis tetapi pemahaman deduksi atas permasalahan (artinya dengan kemampuan induksi-deduksi maka pertanyaan tetap dapat dijawab tanpa mempelajari matematika diskret secara khusus).

C.7. Lain-lain yang relevan dengan potensi akademis

Terdapat sejumlah soal yang memiliki relevansi dalam menguji potensi akademis dan 'computational thinking' yang tidak terkategori dalam tipe-tipe soal di atas.

⁴ Matematika diskret merupakan salah satu cabang ilmu matematika. Soal-soal bertipe C.3 (logika dasar) dan C.4. (aritmatika) sebenarnya termasuk ke dalam bagian ini. Namun berhubung kedua hal tersebut memiliki porsi yang relatif cukup besar maka dalam kisi-kisi ini dikelompokkan secara tersendiri-sendiri.

D. Panduan Pembinaan

D.1 Tingkat OSK

Persiapan

- Seleksi tingkat kabupaten/kota ditujukan untuk menjaring siswa yang mempunyai kemampuan berpikir yang relatif baik. Untuk materi logika, analitika, dan aritmatika, tipe soal yang diujikan serupa dengan soal-soal tes potensi akademik yang sudah cukup dikenal di kalangan akademis. Tidak diperlukan persiapan khusus untuk ketiga materi tersebut, meski latihan soal secara rutin sangat disarankan. Untuk materi algoritmika, peserta sebaiknya mampu membaca (mengerti alur eksekusi) dari sebuah kode/pseudocode program dalam bahasa Pascal karena sebagian algoritma pada soal disajikan dalam bentuk potongan kode/pseudocode.

Bacaan & Referensi

- Seri referensi TOKI dan soal-soal OSK, OSP, OSN TOKI yang bisa diunduh di
 - <http://www.toki.or.id/downloads/>
- Buku-buku Pemrograman Pascal, terutama bagian mengenai:
 - Deklarasi
 - Tipe data
 - Variabel dan konstanta
 - Operasi assignment
 - Operasi aritmetika
 - Ekspresi logika
 - Struktur kontrol
 - Sequence
 - begin—end
 - Analisis kasus (branching)
 - if—then
 - if—then—else
 - case—of
 - Perulangan (loop)
 - while—do
 - repeat—until
 - Fungsi/prosedur
 - Urutan eksekusi
 - Parameter/argumen
 - Rekursivitas
- Bebras Computational Thinking di situs:
 - <http://bebras.or.id>
 - <https://olympia.id>

Contoh Soal

- Lihat Lampiran A

D.2. Tingkat OSP

Persiapan

- Karena pada dasarnya materi uji tingkat provinsi sama dengan materi tingkat kabupaten/kota, tidak diperlukan persiapan khusus untuk seleksi provinsi. Ada baiknya waktu yang ada dipergunakan untuk mulai berlatih membuat program dalam bahasa pemrograman Pascal. Sintaksis dan aturanaturan lain yang spesifik terhadap bahasa Pascal sebaiknya sudah benar-benar dimengerti (dan dihafal) pada tingkat ini.

Bacaan & Referensi

- Sama dengan Tingkat Kabupaten/Kota

Contoh Soal

- Sama dengan Tingkat Kabupaten/Kota

D.3. Tingkat OSN I (Pemrograman Dasar)

Persiapan

- Pada tahap ini, peserta diharapkan sudah dapat merumuskan algoritma sendiri. Untuk mempersiapkan diri, peserta sebaiknya berlatih mengerjakan soal-soal pemrograman yang bersifat problem solving. Mempelajari struktur data dan algoritma-algoritma baku (sorting, searching, dll.) juga disarankan, meski tidak mutlak diperlukan.

Bacaan & Referensi

- USACO Training Program
 - <http://train.usaco.org>
- TOKI Training Gate
 - <https://training.ia-toki.org/>
- Steven Halim, Felix Halim. 2013. *Competitive Programming 3: The New Lower Bound of Programming Contests*.
- Steven Halim. *Data Structures & Algorithm*. CS—NUS.
 - <http://www.comp.nus.edu.sg/~stevenha/programming/programming.html>
- Steven S. Skiena, Miguel Revilla. 2003. *Programming Challenges*. SpringerVerlag.
- Niklaus Wirth. 1976. *Algorithm + Data Structures = Programs*. PrenticeHall.
- Anany Levitin. 2003. *Introduction to the Design and Analysis of Algorithms*. Addison-Wesley.
- Richard Neapolitan, Kumarss Naimipour. 1996. *Foundations of Algorithms*. D.C. Heath and Company.

- Robert Sedgewick. 1988. *Algorithms*. AddisonWesley.
- Steven S. Skiena. 1998. *The Algorithm Design Manual*. Springer Verlag.
- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest. 1989. *Introduction to Algorithms*. The MIT Press.

Contoh Soal

- Lihat Lampiran B

D.4. Tingkat OSN II (Pemrograman Lanjut)

Persiapan

- Pada tahap ini, peserta mulai dihadapkan dengan teknik-teknik desain algoritma dasar dan lanjut. Meski sebagian besar topik akan dibahas sewaktu pelatnas, peserta dianjurkan mempersiapkan diri dengan mempelajari struktur data, algoritma-algoritma baku dan teknik perancangannya. Latihan mengerjakan soal pemrograman secara rutin sangat disarankan untuk menambah “jam terbang”.

Bacaan & Referensi

- Sama dengan Tingkat Nasional I

Contoh Soal

- Lihat Lampiran C
-

Lampiran A

1. Deret bilangan antiFibonacci didefinisikan secara rekursif sbb.:

$$\begin{cases} f_1 = 1 \\ f_2 = 0 \\ f_n = f_{n-2} - f_{n-1} \quad n > 2 \end{cases}$$

Dengan mengambil satu harga n kemudian Anda menjumlahkan bilangan-bilangan tersebut mulai dari 1 s.d. n maka berapakah terkecil agar jumlah itu > 300 ?

- 9
 - 10
 - 14
 - 17
 - 20
2. Jika semua W adalah X, semua X adalah Y, dan semua Y adalah Z, maka manakah yang tidak benar?
- Semua X adalah Z
 - Semua W adalah Y
 - Semua W adalah W
 - Semua W adalah Z
 - Semua Y adalah W
3. Suatu tim peneliti yang terdiri atas lima orang dipilih dari antara empat matematikawan A, B, C, dan D serta empat fisikawan, E, F, G dan H. Paling sedikit tiga matematikawan harus dipilih. Tetapi kesulitannya:
- A tidak mau bekerja dengan D
 - B tidak mau bekerja dengan E
 - F tidak mau bekerja dengan G
 - D tidak mau bekerja dengan F

Jika B dipilih, siapa yang pasti juga akan dipilih?

- F
 - G
 - A
 - C
 - D
4.
$$\begin{array}{r} \text{SEND} \\ \text{MORE +} \\ \hline \text{MONEY} \end{array}$$

S, E, N, D, M, O, R, dan Y masing-masing mewakili satu digit integer (bilangan bulat) positif; dan masing-masing mewakili bilangan yang berbeda. S dan M tidak sama dengan 0.

Berapakah nilai $N+O+S+E$?

- 20
- 16
- 23
- 7
- 11

5. Perhatikan potongan program berikut

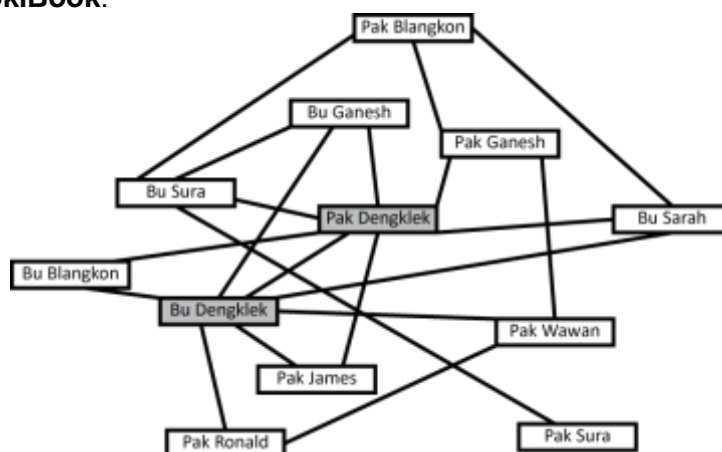
```
n:=10;  
x:=0;  
for i:=0 to n do  
begin  
    x:=x+2*i;  
end;  
writeln(x);
```

Berapakah output dari program di atas?

- 100
 - 112
 - 90
 - 110
 - 72
6. Perhatikan tahapan-tahapan berikut:
Misalkan ada dua variable x dan y , dan variable hasil yang nilai awalnya 0. Lakukan proses berikut hingga nilai x saat ini lebih besar dari 0:
- Jika nilai x genap maka nilai $hasil := hasil + y$.
 - Nilai x selanjutnya adalah nilai x sebelumnya dibagi dua, bila ada hasil pecahan, maka pecahannya dibuang. (contoh bila nilai x sebelumnya 3, maka nilai x selanjutnya 1)
 - Nilai y selanjutnya adalah nilai y sebelumnya dikali tiga

Bila nilai awal $x = 10$ dan nilai awal $y = 5$, maka nilai akhir variable hasil adalah:

- 0
 - 5
 - 25
 - 50
 - Salah semua
7. Ini adalah gambar Pak Dengklek dengan teman-temannya yang terdaftar di media sosial bernama **TokiBook**.



Pada **TokiBook** tersebut, sebuah garis menandakan adanya pertemanan antara dua orang. Pada media tersebut, seseorang dapat mengunggah sebuah foto, *like* sebuah foto, ataupun *share* foto yang diunggahnya. Peraturannya adalah sebagai berikut:

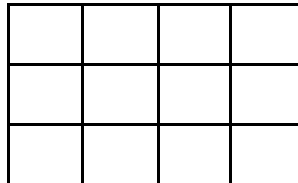
- Seseorang yang mengunggah foto, dapat memilih mau *share* ke teman yang mana, secara spesifik.
- Jika seseorang *me-like* foto anda, seluruh temannya dapat melihat foto anda.

Pak Dengklek ingin mengunggah sebuah foto, tetapi foto tersebut berbahaya apabila dilihat oleh Bu Dengklek. Kepada siapa saja Pak Dengklek dapat meng-*share* fotonya sehingga Bu Dengklek tidak melihat foto tersebut? Tuliskan nama-nama orang yang dapat melihat foto yang dikirim oleh Pak Dengklek dan tidak dapat dilihat oleh bu Dengkle, dipisahkan dengan koma.

Jawaban: **{tuliskan jawaban dalam bentuk string}**

8. MENGHITUNG PERSEGI

Tahukah kamu bahwa grid berukuran 3x4 sebagai berikut memiliki 20 persegi (segi empat sama sisi)?



Diberikan N and M, tentukan banyaknya persegi yang terdapat pada grid berukuran NxM.

Batasan :

$$1 \leq N, M \leq 100.000$$

Format Input :

N M

Format Output :

Sebuah bilangan yang menyatakan banyaknya persegi yang terdapat pada grid berukuran NxM.

Sample input dan output :

3 4	20
4 3	20
1 10	10
5 5	55

Soal:

Buatlah program atau algoritma dari spesifikasi tentang Menghitung Persegi di atas.

Lampiran B

Kata Spiral

Soal OSN 2005

KodeSoal:spiral
Batas Runtime:1 detik / testcase
BatasMemori:1 MB
Masukan:Standard input
Keluaran:Standard output

Deskripsi

Suatu sistem sandi menyandikan kalimat yang diberikan dalam bentuk spiral. Penyusunan tersebut dilakukan membentuk matriks spiral yang dimulai pusat matriks 1 karakter pertama, lalu 1 karakter berikutnya ke kanan, lalu 1 karakter berikutnya ke bawah, lalu 2 karakter berikutnya ke kiri, lalu 2 karakter berikutnya ke atas, 3 karakter berikutnya ke kanan, 3 karakter berikutnya ke bawah, dan seterusnya hingga semua karakter dalam kalimat termasuk dalam spiral. Khususnya, karakter spasi di ganti dengan “_” (underscore), dan jika ada baris/kolom tersisa setelah karakter terakhir maka elemen-elemen matriks diisi juga dengan “_” (underscore) tsb. Misalnya kalimat “Seluruh peserta OSN bidang komputer harus mengerjakan soal-soal sebaikbaiknya untuk mendapatkan peringkat terbaik.” Dikodekan kedalam matriks sebagai berikut.

```
b a i k . _ _ _ _ _  
r a i k n y a _ u n t  
e b m e n g e r j a u  
t - _ b i d a n g k k  
_ k s _ h _ p e _ a _  
t i u N u S e s k n m  
a a r S r u l e o _ e  
k b a O _ a t r m s n  
g e h _ r e t u p o d  
n s _ l a o s - l a a  
i r e p _ n a k t a p
```

Masukan

Program membaca satu baris teks paling panjang 250 karakter.

Keluaran

Program harus menghasilkan sejumlah baris sesuai dengan matriks yang dibentuk. Setiap baris keluaran berisikan karakter-karakter dari baris yang sama berturut-turut dari kolom paling kiri ke paling kanan tanpa pemisahan (karakter-karakter dituliskan bersambung menjadi satu string serta jangan lupa setiap spasi menjadi underscore).

Contoh1

Masukan (tertulis dalam satu baris)

```
Seluruh peserta OSN bidang komputer harus mengerjakan soal-soal  
sebaik-baiknya untuk mendapatkan peringkat terbaik.
```

Keluaran

```
baik._____  
raiknya_unt  
ebmengerjau  
t-_bidangk  
_ks_h_pe_a_  
tiuNuSesknm  
aarSruleo_e  
kba0_atrmsn  
geh_retupod  
ns_laos-laa  
irep_naktap
```

Contoh2

Masukan (tertulis dalam satu baris)

```
TOKI
```

Keluaran

```
TO  
IK
```

Contoh3

Masukan (tertulis dalam satu baris)

```
OSN
```

Keluaran

```
OS  
_N
```

Contoh4

Masukan (tertulis dalam satu baris)

```
Bisakah Kamu?????
```

Keluaran

```
_h_Ka  
_aBim  
_kasu  
?????
```

Lampiran C

Perjamuan Alakapuri

Soal Pelatnas 2006

Nama Soal : Perjamuan Alakapuri

Batasan Memori : 32 MB

Batasan Waktu : 1 s

Input : standard input

Output : standard output

Alkisah, bertujuan untuk memamerkan kekayaannya sekaligus meningkatkan pengaruhnya di kalangan para dewa, dewa kekayaan Kubera berencana mengundang para dewa untuk mengikuti jamuan makan di kotanya, Alakapuri.

Kubera sangat tahu tabiat para dewa yang lain. Para dewa sangat tidak suka diduakan, oleh sebab itu, Kubera harus mengatur jadwal perjamuannya sedemikian rupa sehingga pada suatu saat tertentu, dia hanya menjamu satu dewa saja.

Dengan bantuan para pembantunya, Kubera telah menyusun sebuah daftar yang berisi: nama dewa, waktu di mana dewa tersebut bersedia menghadiri perjamuan, dan perkiraan lama waktu makan dewa tersebut.

Karena Kubera tidak pilih-pilih dalam mengundang para dewa, untuk membuat daftar lebih ringkas, Kubera kemudian berinisiatif mengganti kolom nama dewa menjadi nomor urut saja.

Hasilnya, Kubera mempunyai sebuah tabel yang kira-kira berisi seperti ini:

Dewa	Ke	(i)	Waktu	Awal	Jamuan	(S)	Durasi	(D)
	1				2		5	
	2				9		7	
	3				15		6	
	4				9		3	

Jelas bahwa Kubera mungkin tidak dapat mengundang semua dewa yang ada karena bisa saja jadwal di mana satu dewa bersedia dijamu beririsan dengan jadwal dewa yang lain. (Kubera tidak mungkin mengusir seorang dewa di tengah-tengah perjamuan. Satu perjamuan berakhir ketika dewa yang sedang dijamu selesai makan, dan lama waktu makan dewa tersebut tidak akan lebih lama daripada waktu perkiraan yang telah dibuat pembantu Kubera)

Contohnya, mengacu pada daftar di atas, dia tidak mungkin mengundang keempat dewa yang ada (karena jadwal dewa 2 dan 3 beririsan sehingga hanya salah satu dari mereka yang bisa diundang). Anggap saja, yang diundang adalah dewa 1, 3, dan 4. Dari satuan waktu ke-2 sampai 6 (hingga sesaat sebelum waktu ke-7), Kubera akan menjamu dewa 1. Kemudian pada waktu ke-9, dia akan mulai menjamu dewa 4 sampai waktu ke-11. Dan terakhir, ia menjamu dewa 3 pada waktu 15 – 20.

Sesuai dengan tujuannya semula, Kubera tentu ingin menjamu sebanyak-banyaknya dewa. Sebagai ahli hitung terbaiknya, Anda diminta oleh Kubera untuk pembantunya menghitung berapa banyak dewa yang dapat ia undang ke perjamuannya.

Input

Baris pertama masukan berisi sebuah integer N, menunjukkan banyaknya dewa yang ada. Baris 2, 3, ...,N+1 mendeskripsikan jadwal ke-N dewa tersebut. Baris ke i+1 berisikan dua buah integer Si dan Di, yang merepresentasikan waktu di mana dewa ke-i bisa hadir serta perkiraan durasi perjamuan untuk dewa tersebut.

Output

Terdiri atas sebuah integer M, yaitu banyak maksimum dewa yang dapat dijamu.

Contoh Input

```
4
2 5
9 7
15 6
9 3
```

Contoh Output

```
3
```

Penjelasan Input/Output

Seperti yang diuraikan pada deskripsi soal.

Batasan

$N \leq 100\,000$, $1 \leq S_i \leq 1\,000\,000$, $1 \leq D_i \leq 1\,000$

Catatan: soal ini merupakan soal simulasi pada pelatnas 16 besar TOKI 2006

Disclaimer

Panduan ini dapat berubah sewaktu-waktu.