



Divisi I Pemrograman – Babak Penyisihan

[A] Makan Lumpia

Batas waktu: 1 detik per *test case*

Batas memori: 128 MB

Deskripsi Masalah

Astik baru saja pulang dari liburan di Kota Semarang. Dia membeli oleh-oleh sejumlah lumpia yang terdiri dari A buah lumpia daging, B buah lumpia ayam, C buah lumpia udang, dan D buah lumpia ikan. Setiap harinya dia akan makan lumpia dengan ketentuan sebagai berikut:

1. Setiap harinya Astik akan makan maksimal tiga buah lumpia.
2. Astik boleh memakan lebih dari satu jenis lumpia yang berbeda-beda pada hari yang sama. Namun, lumpia daging dan lumpia ayam tidak boleh dimakan di hari yang sama. Lumpia udang dan lumpia ikan juga tidak boleh dimakan di hari yang sama.
3. Karena setiap jenis lumpia memiliki masa kadaluarsa yang berbeda, dia harus memakan lumpia dengan urutan prioritas yaitu lumpia daging, lumpia ayam, lumpia udang dan lumpia ikan. Ini artinya, jenis lumpia daging harus dihabiskan dulu baru bisa makan lumpia ayam dan seterusnya sesuai urutan prioritas.

Jika diberikan informasi jumlah lumpia pada setiap jenisnya, berapa hari minimal yang dibutuhkan Astik untuk menghabiskan semua lumpia yang dia beli.

Format Masukan dan Keluaran

Masukan terdiri dari satu baris yang berisi empat buah bilangan bulat A , B , C dan D ($0 \leq A, B, C, D \leq 10^9$), yang menyatakan jumlah lumpia daging, lumpia ayam, lumpia udang, dan lumpia ikan berturut-turut. Dijamin Astik membeli minimal 1 lumpia.

Keluaran berupa bilangan bulat yang menyatakan hari minimal yang dibutuhkan Astik untuk menghabiskan semua lumpia.



Divisi I Pemrograman – Babak Penyisihan

Contoh Masukan/Keluaran

Masukan	Keluaran
3 3 3 3	4
2 1 2 1	3
9 0 0 0	3

Penjelasan

Pada contoh pertama, Astik akan memakan tiga buah lumpia tiap harinya pada masing-masing jenis lumpia. Sehingga dia membutuhkan waktu empat hari.

Pada contoh kedua, Astik makan 2 lumpia daging saja pada hari pertama, makan 1 lumpia ayam dan 2 lumpia udang pada hari kedua, dan makan 1 lumpia ikan pada hari ketiga.

Pada contoh ketiga, Astik makan 3 lumpia daging saja selama tiga hari berturut-turut.

[B] Lumpia Rebung

Batas waktu: 1 detik per *test case*

Batas memori: 64 MB

Deskripsi Masalah

Gema dan Astik sedang mengikuti lomba pemrograman di Semarang. Sebelum pulang, Gema membelikan Astik oleh-oleh berupa lumpia rebung sepanjang L cm.

Karena Astik mempunyai N teman, maka Astik ingin memotong lumpia menjadi $N + 1$ bagian (1 bagian untuk dirinya sendiri) dengan memotong N kali. Pada mulanya, lumpia tersebut terdiri atas 1 bagian sepanjang L cm. Pada pemotongan ke- i , Astik memotong bagian ke- B_i dari kiri menjadi 2 bagian sama panjang. Dijamin bagian tersebut mempunyai panjang bilangan genap dalam satuan cm.

Kini Gema penasaran, berapa panjang masing-masing bagian setelah dipotong oleh Astik. Bantulah Gema menghitungnya!

Format Masukan dan Keluaran

Baris pertama masukan terdiri dari dua buah bilangan N ($1 \leq N \leq 100.000$) dan L ($1 \leq L \leq 10^{18}$) yang menyatakan banyaknya teman Astik dan panjang lumpia rebung dalam cm. Baris kedua terdiri dari N buah bilangan yang menyatakan nilai B_1, B_2, \dots, B_N ($1 \leq B_i \leq i$).

Keluaran terdiri dari sebuah baris berisi $N + 1$ bilangan yang menyatakan panjang masing-masing bagian dari kiri sampai kanan dalam cm.

Contoh Masukan/Keluaran

Masukan	Keluaran
4 1000	250 250 125 125 250
1 2 2 1	



Divisi I Pemrograman – Babak Penyisihan

Penjelasan

Pada mulanya, lumpia terdiri dari 1 bagian sepanjang 1000 cm.

Setelah dipotong untuk pertama kalinya, lumpia terdiri dari 2 bagian sepanjang 500 cm dan 500 cm.

Selanjutnya bagian kedua dari kiri dipotong, lumpia terdiri dari 3 bagian sepanjang 500 cm, 250 cm, dan 250 cm.

Kemudian bagian kedua dari kiri dipotong, lumpia terdiri dari 4 bagian sepanjang 500 cm, 125 cm, 125 cm, dan 250 cm.

Terakhir, bagian terkiri dipotong sehingga lumpia terdiri dari 5 bagian sepanjang 250 cm, 250 cm, 125 cm, 125 cm, dan 250 cm.



Divisi I Pemrograman – Babak Penyisihan

[C] Piknik

Batas waktu: 1 detik per *test case*

Batas memori: 128 MB

Deskripsi Masalah

Tahun ini Pak Gema ditugasi menjadi koordinator transportasi pada suatu acara piknik. Ada sebanyak N orang yang akan ikut piknik, Pak Gema harus menyiapkan sejumlah kendaraan yang akan mengangkut semua peserta tersebut. Namun, karena aspirasi dari peserta, Pak Gema harus mengatur sedemikian sehingga semua peserta yang berada pada suatu kendaraan harus merasa nyaman dengan semua peserta lainnya di kendaraan tersebut. Peserta ke- X merasa nyaman dengan peserta ke- Y apabila:

- Peserta ke- X dan peserta ke- Y saling mengenal.
- Peserta ke- X merasa nyaman dengan orang lain yang saling mengenal dengan peserta ke- Y .

Namun, Pak Gema juga perlu memikirkan efisiensi dan tidak mungkin satu kendaraan hanya berisi satu orang. Oleh sebab itu, jika tersisa satu peserta yang tidak mengenal siapapun, maka peserta tersebut akan naik pada kendaraan dengan jumlah peserta paling sedikit. Namun, jika ada lebih dari satu peserta yang tidak mengenal siapapun, maka mereka akan ditempatkan pada kendaraan yang sama.

Jika diberikan informasi tentang peserta mana saja yang saling kenal, bantulah Pak Gema untuk dapat menentukan berapa banyak kendaraan minimal yang harus dia siapkan sehingga semua aturan terpenuhi.

Format Masukan dan Keluaran

Masukan terdiri dari beberapa baris. Baris pertama berisi dua bilangan bulat N ($2 \leq N \leq 100$) dan M ($0 \leq M \leq \frac{N(N-1)}{2}$), yang menyatakan jumlah peserta piknik dan jumlah informasi terkait peserta mana saja yang saling kenal. M baris berikutnya masing-masing terdiri dari dua bilangan bulat berbeda X dan Y ($1 \leq X, Y \leq N, X \neq Y$) yang menyatakan peserta ke- X dan peserta ke- Y saling mengenal satu sama lain. Dijamin tidak ada pasangan yang berulang.

Divisi I Pemrograman – Babak Penyisihan

Keluaran berupa bilangan bulat yang menyatakan banyaknya jumlah kendaraan minimal yang harus disediakan oleh Pak Gema.

Contoh Masukan/Keluaran

Masukan	Keluaran
6 5 1 2 3 1 3 2 4 6 6 5	2
4 3 1 2 2 4 4 3	1
3 1 1 2	1

Penjelasan

Pada contoh pertama, didapatkan bahwa ada dua kelompok yang saling nyaman yang terbentuk, yaitu kelompok pertama yang berisikan peserta ke-1, peserta ke-2, dan peserta ke-3, serta kelompok kedua yang berisikan peserta ke-4, peserta ke-5, dan peserta ke-6. Sehingga Pak Gema harus menyediakan 2 kendaraan.

Pada contoh kedua, semua peserta merasa nyaman satu sama lain, sehingga Pak Gema cukup menyediakan satu kendaraan saja.

Pada contoh ketiga, hanya 2 peserta yang saling mengenal, yaitu peserta ke-1 dan peserta ke-2, sedangkan peserta ke-3 tidak mengenal siapapun. Untuk efisiensi, peserta ke-3 akan ditempatkan pada kendaraan yang sama dengan peserta ke-1 dan peserta ke-2.



Divisi I Pemrograman – Babak Penyisihan

[D] Biner

Batas waktu: 1 detik per *test case*

Batas memori: 128 MB

Deskripsi Masalah

Astik memiliki banyak waktu luang. Karena bingung, ia mencoba menuliskan semua bilangan desimal yang representasi binernya memiliki digit 1 sebanyak **tepat** 30 dari kecil ke besar. Sebagai contoh, representasi biner dari 1073741823 adalah

11111111111111111111111111111111₂

dan memiliki 30 digit 1.

Contoh lainnya, representasi biner dari 68703215615 adalah

111111111110000011111011111111111111₂

yang juga memiliki 30 digit 1.

Anggap bilangan-bilangan yang ditulis Astik adalah X_1, X_2, X_3, \dots , dengan $X_{i+1} > X_i$ untuk setiap bilangan bulat positif i . Tentu $X_1 = 11111111111111111111111111111111_2 = 1073741823$ dan $X_2 = 10111111111111111111111111111111_2 = 1610612735$. Astik ingin tahu berapa nilai desimal dari X_N .

Format Masukan dan Keluaran

Masukan terdiri atas satu buah bilangan bulat positif N ($1 \leq N \leq 10^{15}$).

Keluarkan sebuah bilangan bulat positif yang merupakan nilai desimal dari X_N . Jawaban dijamin muat dalam bilangan bulat 64-bit.

Contoh Masukan/Keluaran

Masukan	Keluaran
1	1073741823
10000000000	1373840604447



Divisi I Pemrograman – Babak Penyisihan

Penjelasan

Pada contoh keluaran pertama, representasi biner dari bilangan 1073741823 adalah

$11111111111111111111111111111111_2$

Pada contoh keluaran kedua, representasi biner dari bilangan 1373840604447 adalah

$10011111111011111010001111111110100011111_2$

Divisi I Pemrograman – Babak Penyisihan

[E] Barisan Fimoracci

Batas waktu: 1 detik per *test case*

Batas memori: 64 MB

Deskripsi Masalah

Astik sangat takjub dengan barisan Fibonacci, sehingga ia pun membuat sebuah barisan baru bernama barisan Fimoracci. Barisan Fimoracci didefinisikan sebagai berikut:

- Barisan terdiri dari **minimal** K buah elemen.
- Barisan dimulai dengan dua buah bilangan bulat: F_1 dan F_2 .
- Untuk $i > 2$, berlaku $F_i \geq F_{i-1} + F_{i-2}$.

Sekarang, Astik sudah memiliki sebuah larik A dengan panjang N yang terurut menaik. Astik ingin mencari tahu berapa banyak subsekuens dari A yang merupakan barisan Fimoracci. Bantulah Astik!

Format Masukan dan Keluaran

Masukan dimulai dengan sebuah baris berisi dua buah bilangan bulat: N dan K ($2 \leq K \leq N \leq 1000$). Baris selanjutnya berisi N buah bilangan bulat A_1, A_2, \dots, A_N ($1 \leq A_1 < A_2 < \dots < A_N \leq 10^9$).

Keluaran terdiri dari sebuah baris berisi banyaknya subsekuens dari A yang merupakan barisan Fimoracci, di-modulo oleh 1.000.000.007.

Contoh Masukan/Keluaran

Masukan	Keluaran
6 3 1 2 3 4 5 6	16
4 4 7 8 9 10	0
10 2 1 2 3 5 8 13 21 34 55 89	1013

Divisi I Pemrograman – Babak Penyisihan

Penjelasan

Pada contoh pertama, terdapat 16 subsekuens dari A yang merupakan barisan Fimoracci:

- [1, 2, 3, 5]
- [1, 2, 3, 6]
- [1, 2, 4, 6]
- [1, 2, 3]
- [1, 2, 4]
- [1, 2, 5]
- [1, 2, 6]
- [1, 3, 4]
- [1, 3, 5]
- [1, 3, 6]
- [1, 4, 5]
- [1, 4, 6]
- [1, 5, 6]
- [2, 3, 5]
- [2, 3, 6]
- [2, 4, 6]

Pada contoh kedua, perhatikan bahwa barisan Fimoracci harus terdiri dari minimal 4 buah elemen dan larik A sendiri bukanlah barisan Fimoracci. Sehingga, jawabannya adalah 0.

Pada contoh ketiga, setiap subsekuens dari A dengan 2 buah elemen atau lebih, dapat dipastikan merupakan barisan Fimoracci. Sehingga, jawabannya adalah $C_2^{10} + C_3^{10} + \dots + C_{10}^{10} = 1013$.

Divisi I Pemrograman – Babak Penyisihan

[F] Serangan Benteng

Batas waktu: 2 detik per *test case*

Batas memori: 256 MB

Deskripsi Masalah

Gema sedang bermain sebuah permainan (*game*) komputer yang baru ia beli. Dalam permainan tersebut, ia perlu menyerang beberapa buah benteng yang dijaga sejumlah tentara. Untuk dapat menguasai sebuah benteng, Gema perlu menyerang dan mengalahkan semua tentara yang menjaga benteng tersebut dengan menggunakan sebuah pesawat tempur. Pesawat tempur tersebut dilengkapi dengan 3 buah jenis senjata:

- Senapan peluru: senjata ini dapat digunakan kapan saja, dan mampu mengenai dan mematikan tepat 1 tentara lawan.
- Bom kecil: hanya bisa digunakan ketika jumlah tentara lawan sebanyak **genap**, dan ketika digunakan dapat mematikan tepat **setengah** dari jumlah tentara lawan yang ada saat itu.
- Bom besar: hanya bisa digunakan ketika jumlah tentara lawan **dapat dibagi tiga**, dan ketika digunakan dapat mematikan tepat **dua pertiga** dari jumlah tentara lawan yang ada saat itu.

Setiap kali akan menggunakan senjata, Gema memerlukan sejumlah koin untuk dapat mengoperasikan senjata tersebut, dengan ketentuan sebagai berikut:

- Senapan peluru: butuh 1 koin setiap kali ditembakkan.
- Bom kecil: butuh P koin setiap kali ditembakkan .
- Bom besar: butuh Q koin setiap kali ditembakkan .

Akan ada N buah babak yang harus diselesaikan oleh Gema. Pada babak ke- i , Gema akan melawan sebanyak A_i tentara. Gema bertanya-tanya, berapa jumlah koin minimal yang diperlukan untuk mengalahkan semua tentara pada setiap babak?

Format Masukan dan Keluaran

Masukan dimulai dengan satu baris yang berisi tiga bilangan bulat positif N ($1 \leq N \leq 100.000$), P , dan Q ($1 \leq P \leq Q \leq 10^9$). N baris berikutnya berisi nilai-nilai A_i ($1 \leq A_i \leq 10^{18}$, $1 \leq i \leq N$), menyatakan banyaknya tentara yang harus dikalahkan Gema pada babak ke- i .

Divisi I Pemrograman – Babak Penyisihan

Keluaran berisi N baris, dengan baris ke- i berisikan satu bilangan bulat yang menunjukkan banyaknya koin minimal yang diperlukan oleh Gema untuk mengalahkan semua tentara lawan pada babak ke- i .

Contoh Masukan/Keluaran

Masukan	Keluaran
2 3 5 16 90	10 18
1 100 100 50	50

Penjelasan

Pada contoh masukan pertama, terdapat dua babak yang akan dihadapi Gema. Biaya pengoperasian bom kecil adalah 3 koin, sedangkan bom besar 5 koin. Pada babak pertama, Gema harus melawan 16 tentara lawan. Untuk mengalahkan 16 tentara ini, Gema dapat melakukan strategi sebagai berikut:

- Jumlah tentara = 16, gunakan bom kecil (biaya 3) sehingga jumlah tentara menjadi 8.
- Sisa jumlah tentara = 8, gunakan bom kecil (biaya 3) sehingga jumlah tentara menjadi 4.
- Sisa tentara (4) dikalahkan dengan menggunakan senapan (masing-masing memerlukan 1 koin, sehingga total = 4 koin).
- Total koin yang diperlukan = $3 + 3 + 4 = 10$.

Dapat dibuktikan bahwa kita tidak dapat mengalahkan lebih dari 16 tentara lawan dengan 10 koin. Oleh karena itu, jawaban pada babak pertama adalah 16. Untuk babak kedua, dapat ditunjukkan dengan cara serupa bahwa untuk mengalahkan 90 tentara lawan, Gema membutuhkan minimal 18 koin.



Divisi I Pemrograman – Babak Penyisihan

[G] *Reseller* Topi Gemastik

Batas waktu: 1 detik per *test case*

Batas memori: 128 MB

Deskripsi Masalah

Gema adalah *reseller* yang sangat handal. Dia bisa mengetahui barang apa yang sedang populer saat ini dan dia akan turut serta menjadi *reseller* barang tersebut.

Dalam bulan ini, Gemastik sedang berlangsung dan tentu Gema ingin menjadi *reseller* barang-barang yang menjadi khas Gemastik, salah satunya adalah topi Gemastik. Saking handalnya, Gema dapat memprediksi harga beli dan harga jual dari satu buah topi Gemastik untuk N hari ke depan. Setelah lewat dari N hari, topi Gemastik sudah tidak populer dan tidak akan ada yang membelinya.

Gema ingin membeli topi Gemastik di salah satu hari tertentu, lalu memasang iklan di keesokan harinya, dan akhirnya menjualnya kembali pada salah satu hari setelahnya. Sebagai contoh, jika Gema membeli topi Gemastik pada hari ke-10, maka pada hari ke-11 ia akan memasang iklan topi Gemastik tersebut, dan penjualan hanya dapat dilakukan pada salah satu hari dari hari ke-12, ke-13, ke-14, dan seterusnya.

Gema meminta bantuan Anda untuk mengetahui apakah Gema bisa memperoleh keuntungan per topi, dan jika iya, berapa keuntungan terbesar yang mungkin diperoleh. Keuntungan adalah selisih antara harga jual dan harga beli, dengan catatan bahwa harga jual lebih tinggi daripada harga beli.

Format Masukan dan Keluaran

Masukan terdiri atas tiga baris. Baris pertama berisikan satu bilangan bulat N ($1 \leq N \leq 100.000$). Baris kedua berisikan N buah bilangan bulat B_1, B_2, \dots, B_N ($1 \leq B_i \leq 10^9$ untuk $1 \leq i \leq N$) yang menyatakan harga beli dari 1 buah topi Gemastik pada hari ke- i . Baris ketiga berisikan N buah bilangan bulat J_1, J_2, \dots, J_N ($1 \leq J_i \leq 10^9$ untuk $1 \leq i \leq N$) yang menyatakan harga jual dari 1 buah topi Gemastik pada hari ke- i .

Keluarkan satu buah bilangan bulat positif yang merupakan keuntungan terbesar per topi yang mungkin diperoleh. Namun, jika Gema tidak mungkin memperoleh keuntungan, maka keluarkan “tidak mungkin”.



Divisi I Pemrograman – Babak Penyisihan

Contoh Masukan/Keluaran

Masukan	Keluaran
6 6 3 6 5 6 3 9 7 9 3 5 8	5
6 9 7 6 8 5 3 6 3 6 5 6 3	tidak mungkin

Penjelasan

Pada Contoh 1, terdapat prediksi harga beli dan harga jual 1 buah topi Gemastik untuk 6 hari ke depan. Salah satu cara untuk mendapatkan keuntungan terbesar adalah dengan membeli topi pada hari ke-2 dengan harga 3, lalu menjualnya pada hari ke-6 dengan harga 8, dan ia memperoleh keuntungan sebesar 5. Perhatikan bahwa jika Gema membeli topi Gemastik pada hari ke-2, dia baru bisa menjual topi Gemastik mulai hari ke-4.

Pada Contoh 2, terdapat prediksi harga beli dan harga jual 1 buah topi Gemastik untuk 6 hari ke depan. Dapat dilihat bahwa pada hari apapun ketika Gema membeli topi Gemastik, dia tidak akan bisa memperoleh keuntungan. Oleh karena itu, keluarannya adalah "tidak mungkin".



Divisi I Pemrograman – Babak Penyisihan

[H] Tumpukan Kartu

Batas waktu: 1 detik per *test case*

Batas memori: 64 MB

Deskripsi Masalah

Gema dan Astik mempunyai tumpukan kartu yang awalnya berisi N buah kartu. Kartu ke- i dari atas bertuliskan angka A_i . Dijamin bahwa seluruh kartu dalam tumpukan bertuliskan angka yang berbeda-beda.

Mereka akan melakukan K operasi pada tumpukan kartu, yang masing-masing adalah salah satu dari berikut:

- Astik menambahkan kartu bertuliskan angka X ke atas tumpukan kartu. Dijamin bahwa kartu tersebut sebelumnya tidak ada pada tumpukan kartu tersebut. Sebagai contoh, jika tumpukan kartu bertuliskan angka 1,2,3,4 dan $X = 10$, maka tumpukan kartu akan menjadi 10,1,2,3,4.
- Gema mengambil kartu bertuliskan angka Y dan semua kartu di bawahnya, lalu meletakkannya di atas tumpukan kartu tanpa mengubah urutan. Sebagai contoh, jika tumpukan kartu bertuliskan angka 1,2,3,4 dan $Y = 3$, maka tumpukan kartu akan menjadi 3,4,1,2. Jika ternyata kartu dengan angka Y terletak pada posisi paling atas, maka Gema tidak akan melakukan apapun.

Anda diberikan kondisi awal tumpukan dan urutan operasi yang akan mereka lakukan. Tugas Anda adalah mengeluarkan kondisi akhir tumpukan setelah semua operasi dilakukan.

Format Masukan dan Keluaran

Baris pertama masukan terdiri dari dua buah bilangan N dan K ($1 \leq N, K \leq 100.000$) yang menyatakan banyaknya kartu pada tumpukan awal dan banyaknya operasi. Baris kedua terdiri dari N buah bilangan yang menyatakan nilai A_1, A_2, \dots, A_N ($1 \leq A_i \leq 10^9$). K baris berikutnya berbentuk salah satu dari dua format berikut:

- A <X>
Astik menambahkan kartu bertuliskan X ($1 \leq X \leq 10^9$) ke atas tumpukan kartu. Dijamin bahwa tidak ada kartu bertuliskan X dalam tumpukan sebelum operasi ini.

Divisi I Pemrograman – Babak Penyisihan

- G <Y>

Gema mengambil kartu bertuliskan Y ($1 \leq Y \leq 10^9$) dan semua kartu di bawahnya, lalu meletakkannya di atas tumpukan kartu tanpa mengubah urutan. Dijamin bahwa ada kartu bertuliskan Y dalam tumpukan ketika operasi ini.

Baris pertama keluaran berisi sebuah bilangan yang menyatakan banyaknya kartu dalam tumpukan setelah semua operasi dilakukan. Baris kedua berisi angka-angka yang tertulis pada kartu-kartu dalam tumpukan dari kartu paling atas ke kartu paling bawah.

Contoh Masukan/Keluaran

Masukan	Keluaran
5 3 1 2 3 4 5 G 3 A 10 G 4	6 4 5 1 2 10 3

Penjelasan

Setelah Gema mengambil kartu bertuliskan angka 3, tumpukan kartu menjadi 3, 4, 5, 1, 2.

Setelah Astik menambahkan kartu bertuliskan angka 10, tumpukan kartu menjadi 10, 3, 4, 5, 1, 2.

Setelah Gema mengambil kartu bertuliskan angka 3, tumpukan kartu menjadi 4, 5, 1, 2, 10, 3.



Divisi I Pemrograman – Babak Penyisihan

[I] Diskon 50%, S&K Berlaku

Batas waktu: 1 detik per *test case*

Batas memori: 128 MB

Deskripsi Masalah

Di sebuah toko buku, sedang ada promo untuk Q hari ke depan. Diketahui pada hari ke- i , Gema dapat memperoleh promo dengan syarat dan ketentuan sebagai berikut:

- Dalam satu buah transaksi, Gema hanya boleh membeli maksimal X_i buah buku.
- Untuk setiap transaksi yang mana Gema membeli tepat X_i buah buku, maka satu buku **termurah** yang dibeli di transaksi tersebut akan mendapatkan diskon 50% dengan **pembulatan ke bawah**. Sebagai contoh, jika harga dari buku yang didiskon adalah 7, maka diskonnya adalah 3, sehingga harganya menjadi 4.
- Gema boleh melakukan sebanyak apa pun transaksi pada hari tersebut.

Gema ingin berbelanja N buah buku yang dinomori dari 1 hingga N , dengan harga buku ke- j adalah A_j . Gema ingin tahu untuk setiap hari ke- i ($1 \leq i \leq Q$), berapa biaya termurah yang mungkin seandainya ia membeli seluruh N buku pada hari ke- i tersebut?

Format Masukan dan Keluaran

Masukan dimulai dengan sebuah baris berisi dua buah bilangan bulat: N dan Q ($1 \leq N, Q \leq 200.000$). Baris selanjutnya berisi N buah bilangan bulat A_1, A_2, \dots, A_N ($1 \leq A_j \leq 10^9$ untuk $1 \leq j \leq N$). Q baris selanjutnya masing-masing berisi sebuah bilangan bulat X_i ($2 \leq X_i \leq 200.000$).

Keluaran terdiri dari Q baris dengan baris ke- i berisi sebuah bilangan bulat yang menyatakan biaya termurah yang mungkin untuk membeli seluruh N buku pada hari ke- i .

Divisi I Pemrograman – Babak Penyisihan

Contoh Masukan/Keluaran

Masukan	Keluaran
6 3	270
7 50 99 50 50 42	277
3	270
5	
3	

Penjelasan

Untuk hari ke-1 dan juga hari ke-3, Gema memperoleh biaya termurah dengan melakukan transaksi-transaksi sebagai berikut:

- **Transaksi 1:**
 - Membeli buku ke-2, ke-4, dan ke-5 dengan harga 50, 50, dan 50.
 - Buku termurah memiliki harga 50, sehingga didiskon 50% menjadi 25.
 - Total harga menjadi $50 + 50 + 25 = 125$.
- **Transaksi 2:**
 - Membeli buku ke-1, ke-3, dan ke-6 dengan harga 7, 99, dan 42.
 - Buku termurah memiliki harga 7, sehingga setelah diskon 50% (dengan pembulatan ke bawah), harganya menjadi 4.
 - Total harga menjadi $4 + 99 + 42 = 145$.

Biaya yang dikeluarkan Gema adalah $125 + 145 = 270$.

Untuk hari ke-2, Gema memperoleh biaya termurah dengan melakukan transaksi-transaksi sebagai berikut:

- **Transaksi 1:**
 - Membeli buku ke-2 hingga ke-6 dengan harga 50, 99, 50, 50, dan 42.
 - Buku termurah memiliki harga 42, sehingga didiskon 50% menjadi 21.
 - Total harga menjadi $50 + 99 + 50 + 50 + 21 = 270$.
- **Transaksi 2:**
 - Membeli buku ke-1 dengan harga 7.
 - Tidak ada diskon pada transaksi ini sehingga total harga tetaplah 7.

Biaya yang dikeluarkan Gema adalah $270 + 7 = 277$.



Divisi I Pemrograman – Babak Penyisihan

[J] Tupai Loncat

Batas waktu: 1 detik per *test case*

Batas memori: 128 MB

Deskripsi Masalah

Gema dan Astik memiliki seekor binatang peliharaan kesayangan berjenis tupai loncat, yang mereka beri nama Tika. Gema dan Astik ingin agar Tika menjadi sehat dan semakin lincah. Untuk itu, mereka sering melatih Tika melakukan lompatan-lompatan. Cara mereka melakukan latihan untuk Tika adalah dengan menyediakan sederetan tiang, kemudian meminta Tika untuk loncat dari satu tiang ke tiang berikutnya. Tika harus melakukan lompatan dari tiang pertama sampai dengan tiang terakhir. Tiang-tiang tersebut mungkin memiliki ketinggian yang berbeda-beda. Bagi Tika, tingkat kesulitan untuk melakukan lompatan dari satu tiang ke tiang berikutnya adalah sebesar selisih antara ketinggian dari kedua tiang tersebut.

Dalam sebuah sesi latihan, akan ada N buah tiang dengan tiang ke- i memiliki ketinggian a_i . Gema dan Astik kebetulan memiliki pendekatan yang berbeda dalam melatih Tika. Gema lebih santai dan tidak terlalu memaksakan terhadap Tika untuk berlatih. Oleh karena itu, ketika ia menyusun N buah tiang-tiang tersebut, ia akan menyusunnya sedemikian rupa sehingga tingkat kesulitan terbesar dari lompatan yang harus dilakukan Tika adalah sekecil mungkin. Sebaliknya, Astik memiliki pendekatan yang lebih keras dalam melatih Tika. Oleh karena itu, ketika ia menyusun tiang-tiang tersebut, ia akan selalu membuat sedemikian rupa sehingga tingkat kesulitan terbesar dari lompatan yang harus dilakukan Tika adalah sebesar mungkin.

Bantulah Tika untuk bersiap-siap menghadapi sesi latihan yang harus ia lakukan, dengan cara menghitung berapa berapa tingkat kesulitan terbesar dari lompatan yang harus ia lakukan, baik ketika latihan tersebut ditentukan oleh Gema maupun oleh Astik.

Format Masukan dan Keluaran

Masukan terdiri atas dua baris. Baris pertama berisi sebuah bilangan bulat N ($2 \leq N \leq 100.000$) menyatakan banyaknya tiang yang akan digunakan pada sesi latihan. Baris kedua terdiri atas N bilangan bulat positif a_1, a_2, \dots, a_N ($1 \leq a_i \leq 10^9$) menyatakan tinggi dari tiang-tiang tersebut.

Divisi I Pemrograman – Babak Penyisihan

Keluaran terdiri dari dua buah baris, masing-masing berisi sebuah bilangan bulat. Bilangan pertama adalah tingkat kesulitan terbesar dari loncatan yang harus dilakukan oleh Tika, apabila yang mengatur urutan tiang-tiang adalah Gema. Bilangan kedua adalah tingkat kesulitan terbesar dari loncatan yang harus dilakukan oleh Tika, apabila yang mengatur urutan tiang-tiang adalah Astik.

Contoh Masukan/Keluaran

Masukan	Keluaran
3	3
3 1 6	5
4	1
2 2 1 1	1

Penjelasan

Pada contoh kasus pertama, ada tiga tiang dengan tinggi masing-masing adalah 3, 1 dan 6. Gema dapat mengatur urutan tiang menjadi 6, 3, 1, sehingga tingkat kesulitan loncatan tertinggi akan terjadi antara tiang pertama dan kedua (6 dan 3) dengan nilai tingkat kesulitan = $|6 - 3| = 3$. Tidak ada urutan lain yang menyebabkan tingkat kesulitan terbesar bernilai kurang dari 3. Sebaliknya, Astik dapat mengatur urutan tiang menjadi 6, 1, 3, sehingga tingkat kesulitan loncatan tertinggi akan terjadi antara tiang pertama dan kedua (6 dan 1) dengan nilai tingkat kesulitan = $|6 - 1| = 5$. Tidak ada urutan lain yang menyebabkan tingkat kesulitan terbesar bernilai lebih dari 5.