

1. Special Price (T0901)

Di suatu kota ada sebuah toko baru yang merupakan toko barang antik dan juga toko grosir yang unik bernama Special Price. Toko grosir pada umumnya menerapkan aturan semakin banyak jumlah barang sejenis yang dibeli semakin murah harganya. Namun karena toko Special Price juga merupakan toko barang antik, aturan yang diterapkan sedikit berbeda, yaitu:

1. Satu jenis barang hanya bisa dibeli sebanyak satu buah saja.
2. Jumlah jenis barang maksimum yang boleh dibeli ditentukan oleh pemilik toko.
3. Setiap barang memiliki harga masing-masing, namun total harga dihitung dengan rumus:

$$(a_1 + a_2 + a_3 \dots + a_n)/x$$

dengan:

a_1, a_2, \dots, a_n adalah harga barang 1, harga barang 2, ..., harga barang n ,

n adalah jumlah jenis barang maksimum yang boleh dibeli,

dan x adalah jumlah total barang dimiliki toko Special Price.

Karena masih baru, toko Special Price belum mempunyai banyak barang antik untuk dijual. Jumlah barang antik bertambah satu buah tiap harinya. Pada hari pertama toko Special Price hanya memiliki satu barang antik.

Jika jumlah jenis barang maksimum yang boleh dibeli sebanyak n , pemilik toko ingin mengetahui berapa hasil penjualan minimum tiap harinya mulai dari hari ke- n setelah tokonya buka dengan asumsi barang yang terjual pada hari itu sebanyak n buah dan tidak ada barang yang terjual di hari-hari sebelumnya. Buatlah program untuk membantu pemilik toko Special Price tersebut dengan menggunakan Priority Queue yang diimplementasikan dengan Heap buatan anda sendiri.

Contoh:

Maksimum jumlah jenis barang yang boleh dibeli = 3

Harga barang baru di hari ke-1 = 8	Pendapatan minimal pada hari ke-3 = $(8 + 7 + 6)/3 = 7.00$
Harga barang baru di hari ke-2 = 7	Pendapatan minimal pada hari ke-4 = $(8 + 7 + 6)/4 = 5.25$
Harga barang baru di hari ke-3 = 6	Pendapatan minimal pada hari ke-5 = $(7 + 6 + 1)/5 = 2.80$
Harga barang baru di hari ke-4 = 9	Pendapatan minimal pada hari ke-6 = $(6 + 4 + 1)/6 = 1.83$
Harga barang baru di hari ke-5 = 1	Pendapatan minimal pada hari ke-7 = $(4 + 3 + 1)/7 = 1.14$
Harga barang baru di hari ke-6 = 4	
Harga barang baru di hari ke-7 = 3	

Spesifikasi Input

Input diawali dengan dua buah bilangan bulat K ($0 \leq K \leq 10.000$) dan N ($1 \leq N \leq K$), yang menunjukkan jumlah barang yang tersedia dan jumlah maksimum barang yang boleh dibeli. Nilai N selalu lebih kecil dari K . K baris berikutnya masing-masing berisi sebuah bilangan bulat A_i ($0 \leq A_i \leq 100.000$) yang menunjukkan harga barang ke- i .

Spesifikasi output

Untuk setiap harinya, mulai dari hari ke- N hingga ke- K , keluarkan sebuah bilangan pecahan dengan ketelitian dua angka dibelakang koma yang menunjukkan penjualan minimum jika terjual N buah barang pada hari itu.

Contoh Input

7 3
8
7
6
9
1
4
3

Contoh Output

7.00
5.25
2.80
1.83
1.14