

Started on Thursday, 7 April 2022, 11:00 AM

State Finished

Completed on Thursday, 7 April 2022, 1:13 PM

Time taken 2 hours 13 mins

Marks 292.00/300.00

Grade 97.33 out of 100.00

Question **1**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Buatlah sebuah kelas dengan nama **Utility**. Di dalam kelas tersebut terdapat berbagai method **generic** yang bersifat statik yang dapat digunakan untuk mengelola sebuah array satu dimensi. Method-method tersebut adalah:

1. **swap**, digunakan untuk menukar 2 elemen pada sebuah array (tidak menghasilkan array baru). Terdapat 3 parameter yaitu array generik, index *i*, dan index *j*. Apabila index yang ingin ditukar tidak terdapat pada array, maka array tidak berubah.
2. **repeat**, digunakan untuk membuat array baru yang merupakan pengulangan sebuah array sebanyak *N* kali. Terdapat 2 parameter yaitu array generik dan integer positif *N*.
3. **join**, digunakan untuk menggabungkan seluruh elemen pada array menjadi sebuah string dengan delimiter `" "`. Terdapat 1 parameter yaitu array generik. Contoh: `[1,2,3]` menjadi `"1,2,3"`. Perhatikan bahwa tipe data lainnya akan berubah menjadi string yang sesuai, misal `true` menjadi `"true"`, `null` menjadi `"null"`, dst.
4. **sum**, digunakan untuk menghasilkan jumlah dari seluruh elemen array dengan tipe double. Terdapat 1 parameter yaitu array generik. Method hanya bisa digunakan untuk array dengan tipe elemen berupa **Number**.

Kumpulkan file dengan nama `Utility.java`

Java 8

 [Utility.java](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	25	Accepted	0.27 sec, 28.46 MB
2	25	Accepted	0.45 sec, 27.92 MB
3	25	Accepted	0.41 sec, 29.01 MB
4	25	Accepted	0.25 sec, 28.56 MB

Question **2**

Correct

Mark 100.00 out of 100.00

Time limit

1 s

Memory limit

64 MB

Morjgeen Schudjanto adalah seorang alchemist di tahun 1706 yang berasal dari Batavia Utara, Hindia Belanda. Beliau merupakan orang yang menciptakan benda yang disebut dengan **NummerReageerbuisje**. NummerReageerbuisje adalah benda yang unik, mirip dengan test tube di era modern namun memudahkan angka-angka diskrit. Angka-angka yang masuk ke dalam NummerReageerbuisje akan tertumpuk layaknya sebuah stack, sehingga saat NummerReageerbuisje dituangkan maka angka yang keluar lebih dulu adalah angka yang memiliki posisi paling atas.

Tugas Anda adalah mensimulasikan perilaku NummerReageerbuisje ciptaan Morjgeen Schudjanto dengan kelas **Stack** dari Java API. Lengkapi kelas **NummerReageerbuisje** dari berkas [NummerReageerbuisje.java](#) yang diberikan sebagai template.

Kumpulkan **NummerReageerbuisje.java**

Hint: Anda boleh menggunakan **ArrayList** atau fungsi dari Java **Collections**

Java 8

 [NummerReageerbuisje.java](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	12	Accepted	0.17 sec, 28.38 MB
2	12	Accepted	0.14 sec, 27.79 MB
3	12	Accepted	0.14 sec, 27.81 MB
4	12	Accepted	0.13 sec, 28.05 MB
5	12	Accepted	0.16 sec, 28.13 MB
6	12	Accepted	0.13 sec, 26.32 MB
7	12	Accepted	0.12 sec, 26.82 MB
8	16	Accepted	0.09 sec, 30.89 MB

Question **3**
Partially correct
Mark 92.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Problem Statement

Terdapat N buah pekerjaan yang dapat dikerjakan oleh M pekerja secara paralel. Kalian diminta untuk menghitung waktu yang diperlukan untuk menyelesaikan pekerjaan tersebut berdasarkan durasi pengerjaan masing-masing pekerjaan. Pekerjaan dikerjakan secara terurut.

Lengkapi [Worker.java](#) yang diberikan untuk menghitung waktu yang diperlukan untuk menyelesaikan seluruh pekerjaan!

Hint: manfaatkan Java API List, Java Collections, dan Java Stream API untuk mempermudah pengerjaan soal ini!

Sample Input & Output

Contoh 1:

```
N=5
M=1
Durasi=5,1,3,2,4
```

Dengan satu pekerja saja, masing-masing pekerjaan harus dikerjakan satu-persatu. Sehingga, diperlukan $5+1+3+2+4 = 15$ satuan waktu.

Contoh 2:

```
N=5
M=2
Durasi=5,1,3,2,4
```

Dengan dua pekerja dapat dikerjakan dua pekerjaan sekaligus. Bila disimulasikan, akan dibutuhkan 9 satuan waktu saja. Perlu diperhatikan bahwa pekerjaan masuk secara terurut berdasarkan masukan. Berikut adalah simulasinya:

1. Pekerjaan dengan durasi 5 dan 1 dikerjakan.
2. Pekerjaan dengan durasi 5 masih dikerjakan (sisa 4 satuan waktu), pekerjaan dengan durasi 3 mulai dikerjakan. (+1)
3. Pekerjaan dengan durasi 5 masih dikerjakan (sisa 1 satuan waktu), pekerjaan dengan durasi 2 mulai dikerjakan. (+3)
4. Pekerjaan dengan durasi 4 mulai dikerjakan, pekerjaan dengan durasi 2 masih dikerjakan (sisa 1 satuan waktu). (+1)
5. Sudah tidak ada pekerjaan yang belum dikerjakan, sisa pekerjaan dikerjakan sampai habis. (+4)

Sehingga, diperlukan 9 satuan waktu.

Constraints

```
1<=N<=1000
1<=M<=100
1<=Pi<=1000000, dengan Pi durasi pengerjaan untuk pekerjaan ke-i
```

Java 8

 [Worker.java](#)

Score: 92

Blackbox

Score: 92

Verdict: Time limit exceeded

Evaluator: Exact

No	Score	Verdict	Description
1	8	Accepted	0.54 sec, 33.67 MB
2	8	Accepted	0.70 sec, 33.20 MB
3	0	Time limit exceeded	1.04 sec, 35.46 MB
4	8	Accepted	0.86 sec, 37.50 MB
5	8	Accepted	0.80 sec, 33.39 MB

5	8	Accepted	0.88 sec, 33.53 MB
No	Score	Verdict	Description

6	8	Accepted	0.39 sec, 29.91 MB
7	8	Accepted	0.99 sec, 33.42 MB
8	8	Accepted	0.72 sec, 33.57 MB
9	8	Accepted	0.76 sec, 33.39 MB
10	8	Accepted	0.83 sec, 33.24 MB
11	8	Accepted	0.77 sec, 33.22 MB
12	12	Accepted	0.97 sec, 33.24 MB

[◀ Q&A Minggu 12](#)

Jump to...