Started on	Friday, 20 June 2025, 1:00 PM
State	Finished
Completed on	Friday, 20 June 2025, 2:49 PM
Time taken	1 hour 48 mins
Grade	**380.00** out of 400.00 (**95**%)

Question **1**

Partially correct

Mark 80.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Aplikasi Sketsa

Rian, seorang desainer digital, sedang merancang aplikasi sketsa versinya sendiri. Tujuannya adalah membuat aplikasi yang tidak hanya bisa menggambar, tetapi dapat disesuaikan dengan berbagai kebutuhannya jika diperlukan. Setiap aksi yang dilakukan Rian, seperti menggambar bentuk atau menambahkan catatan, dicatat oleh aplikasi sebagai sebuah "Command".

Dua perintah utama adalah:

1. **`DrawShapeCommand`**: Saat Rian menggambar sebuah objek.
2. **`LogMessageCommand`**: Saat Rian menambahkan catatan teks pada kanvasnya.

Selain itu, terdapat beberapa file lainnya yaitu

1. **Command** sebagai antarmuka/interface untuk implementasi dari command
2. **InvalidCommandException** untuk mengurus kesalahan program dengan kondisi tertentu
3. **CommandProcessor** kelas utama untuk pemrosesan command

Kumpulkan **CommandProcessor.java, DrawShapeCommand.java, InvalidCommandException.java , dan LogMessageCommand.java** dalam file **Command.zip**

Java 8

Command.zip

Score: 80

Blackbox

Score: 80

Verdict: Wrong answer

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.07 sec, 29.10 MB
2	20	Accepted	0.06 sec, 27.90 MB
3	20	Accepted	0.06 sec, 28.89 MB
4	20	Accepted	0.07 sec, 28.83 MB
5	0	Wrong answer	0.07 sec, 28.75 MB

Question **2**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Di sebuah negeri perbentukan, terdapat sebuah pabrik. Dalam dunia ini, mereka hanya mengenal CIRCLE, TRIANGLE, dan RECTANGLE.
Menggunakan Factory Pattern, implementasikan ShapeFactory.java. Berikut Shape.java (masukkan ke dalam zip)

Hint: Klik semua link biru untuk mendapatkan file-file yang perlu diimplementasikan, kemudian setelah selesai diimplementasikan semua filenya di zip dalam satu zip file (hati-hati, bukan folder yang di zip ya!). File-file yang ada di dalam zip adalah sbb: Circle.java, Triangle.java, Rectangle.java, dan ShapeFactory.java

Kumpulkan dalam Answer.zip

Java 8

Shape.zip

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.06 sec, 26.66 MB
2	20	Accepted	0.08 sec, 26.20 MB
3	20	Accepted	0.11 sec, 28.13 MB
4	20	Accepted	0.07 sec, 28.93 MB
5	20	Accepted	0.07 sec, 28.48 MB

Question **3**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Average/Min/Max Paralel dengan Java Thread

Deskripsi Singkat

Dengan memanfaatkan **`Thread`**, tugas perhitungan **rata-rata (mean)**, pencarian **nilai minimum**, dan **nilai maksimum** dari elemen array dapat dibagi ke beberapa unit kerja agar dapat dieksekusi secara paralel. Hasil akhir kemudian akan diagregasikan dari setiap *partial result* yang dikerjakan oleh masing-masing *thread*.

Spesifikasi Soal

Diberikan file `Main.java` berisi kerangka program dengan bagian `// TODO` yang belum diisi

Tugas Anda

Lengkapi dua bagian yang masih ber-`TODO` di `Main.java`:

1. Pada `main`

- Implementasikan logika untuk membagi array **`data`** menjadi **`T`** segmen sesuai aturan pembagian yang diberikan dalam komentar.
- Untuk setiap segmen, buat dan kelola eksekusi sebuah SumMinMaxThread agar dapat berjalan secara paralel.
- Pastikan program menunggu hingga semua thread selesai sebelum melanjutkan ke proses agregasi hasil akhir.

2. Pada `SumMinMaxThread.run()`

- Dalam metode `run()`, hitung `sum, min, max` dari segmen array yang ditugaskan kepada thread.
- Simpan hasil perhitungan lokal ini ke dalam array `partialSum`, `partialMin`, dan `partialMax` pada indeks yang sesuai.

Contoh input:

```
8
1 2 3 4 5 6 7 8
3
```

Contoh output:

```
Minimum = 1
Maximum = 8
Mean    = 4.50
```

Kumpulkan file `Main.java` yang sudah menyelesaikan kedua TODO di atas.

Java 8

Main.java

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.11 sec, 29.04 MB
2	20	Accepted	0.08 sec, 27.99 MB
3	20	Accepted	0.08 sec, 28.45 MB
4	20	Accepted	0.08 sec, 27.05 MB
5	20	Accepted	0.08 sec, 28.83 MB

Question **4**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Reflection Calculator

Deskripsi Singkat

Anda akan mengimplementasikan sebuah program yang memanfaatkan **Java Reflection** untuk memanggil metode dan memanipulasi *field* dari sebuah kelas `Calculator`. Tujuannya adalah untuk melakukan operasi aritmatika (penjumlahan, pengurangan, perkalian, pembagian) dan mengubah presisi pembagian secara dinamis menggunakan *reflection*.

Spesifikasi Soal

Anda diberikan dua file Java:

1. `Calculator.java`: Kelas yang berisi logika operasi aritmatika.
2. `ReflectionCalculator.java`: Kerangka program yang akan Anda lengkapi untuk memanggil metode `Calculator` menggunakan *reflection*.

Tugas Anda

Lengkapi `ReflectionCalculator.java`:

- Untuk **`tc = 1`** (penjumlahan): Gunakan *reflection* untuk memanggil metode `add` dari objek `Calculator`.
- Untuk **`tc = 2`** (pengurangan): Gunakan *reflection* untuk memanggil metode `subtract` dari objek `Calculator`.
- Untuk `tc = 3` (perkalian): Gunakan *reflection* untuk memanggil metode `multiply` dari objek `Calculator`.
- Untuk `tc = 4` (pembagian dengan presisi default): Gunakan *reflection* untuk memanggil metode `divide` dari objek `Calculator`.
- Untuk `tc = 5` (mengatur presisi menjadi 3): Gunakan *reflection* untuk mengubah nilai *field* `precision` di objek `Calculator` menjadi `3`, kemudian panggil kembali metode `divide` dengan *reflection* untuk menampilkan hasil dengan presisi yang baru.

Contoh input:

```
1
10 5
```

Contoh output:

```
add      (10, 5) = 15
```

Kumpulkan file `ReflectionCalculator.java` yang sudah menyelesaikan kedua TODO di atas.

Java 8

ReflectionCalculator.java

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	20	Accepted	0.09 sec, 29.00 MB
2	20	Accepted	0.09 sec, 28.02 MB
3	20	Accepted	0.09 sec, 28.97 MB
4	20	Accepted	0.09 sec, 29.92 MB
5	20	Accepted	0.09 sec, 26.41 MB

◄ Praktikum 6 (Latihan)

Jump to...