

Instruction

1. Import the whole project “Scheduling System” or the src code to a proper Java IDE, like Eclipse.
2. Run Main.java. A sentence “The CPU scheduler is working...” in Console and a GUI “Read Keyboard Input” will appear.
3. For the GUI, each row represents one job to run. Enter numbers in these three input boxes in a row, and make sure to press Return key after each input. After pressing the last Return key, the job will run in Console. That is, count itself from the InitialValue to the EndValue.
4. Finish the rest three rows in the GUI and enjoy the output in Console.

The output result should be related to the arrival time of each job, which depends on the input speed in the GUI. The following examples partly show the relation between input speed and output result.

Example 1:

Finish input of all four jobs in one quantum, which is the time to count four numbers in Console.

The first three jobs alternately run until one of them is completed, then JOB 4 begins to run.

The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Shows the project structure with packages like 'LeetCode', 'Scheduling System', 'src', 'jobs', 'main', 'roundrobin', 'threads', and 'JRE System Library'.
- Main.java:** Contains the following code:

```
1 package main;
2
3 import threads.*;
4
5 public class Main {
6     public static void main(String[] args) {
7         // Multithreading "CPU" & "Task" run
8         CPU.getInstance().start();
9
10        new Timer();
11
12        new ReadKeyboard();
13    }
14 }
15
16
17
18
```
- Read Keyboard Input Dialog:** A modal dialog with the title 'Read Keyboard Input'. It contains a table with 4 rows and 3 columns: 'JOB No.', 'Initial Value', and 'End Value'. The values are as follows:

JOB No.	Initial Value	End Value
1	1	12
2	1	9
3	1	8
4	1	5
- Console:** Displays the output of the program, showing the execution of the scheduler and the running status of the jobs. The output is as follows:

```
Main [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_45.jdk/
The CPU scheduler is working...

JOB 1 is running... [ 1 time(s) ]
JOB 1 is running... [ 2 time(s) ]
JOB 1 is running... [ 3 time(s) ]
JOB 1 is running... [ 4 time(s) ]
JOB 2 is running... [ 1 time(s) ]
JOB 2 is running... [ 2 time(s) ]
JOB 2 is running... [ 3 time(s) ]
JOB 2 is running... [ 4 time(s) ]
JOB 3 is running... [ 1 time(s) ]
JOB 3 is running... [ 2 time(s) ]
JOB 3 is running... [ 3 time(s) ]
JOB 3 is running... [ 4 time(s) ]
JOB 1 is running... [ 5 time(s) ]
JOB 1 is running... [ 6 time(s) ]
JOB 1 is running... [ 7 time(s) ]
JOB 1 is running... [ 8 time(s) ]
JOB 2 is running... [ 5 time(s) ]
JOB 2 is running... [ 6 time(s) ]
JOB 2 is running... [ 7 time(s) ]
JOB 2 is running... [ 8 time(s) ]
JOB 3 is running... [ 5 time(s) ]
JOB 3 is running... [ 6 time(s) ]
JOB 3 is running... [ 7 time(s) ]
JOB 3 is running... [ 8 time(s) ]
```

Java - Scheduling System/src/main/Main.java - Eclipse - /Users/Suming/workspace

Package Explorer: LeetCode, Scheduling System, src, jobs, AbstractJob.java, CounterJob.java, main, Main.java, roundrobin, Scheduler.java, threads, CPU.java, ReadKeyboard.java, Task.java, Timer.java, JRE System Library

Main.java

```
1 package main;
2
3 import threads.*;
4
5 public class Main {
6     public static void main(String[] args) {
7         // Multithreading "CPU" & "Task" run
8         CPU.getInstance().start();
9
10        new Timer();
11
12        new ReadKeyboard();
13
14    }
15 }
16
17
18
```

Read Keyboard Input

JOB No.	Initial Value	End Value
1	1	12
2	1	9
3	1	8
4	1	5

Console

Main [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_45.jdk/

JOB 1 is running... [5 time(s)]

JOB 1 is running... [6 time(s)]

JOB 1 is running... [7 time(s)]

JOB 1 is running... [8 time(s)]

JOB 2 is running... [5 time(s)]

JOB 2 is running... [6 time(s)]

JOB 2 is running... [7 time(s)]

JOB 2 is running... [8 time(s)]

JOB 3 is running... [5 time(s)]

JOB 3 is running... [6 time(s)]

JOB 3 is running... [7 time(s)]

JOB 3 is running... [8 time(s)]

JOB 1 is running... [9 time(s)]

JOB 1 is running... [10 time(s)]

JOB 1 is running... [11 time(s)]

JOB 1 is running... [12 time(s)]

JOB 4 is running... [1 time(s)]

JOB 4 is running... [2 time(s)]

JOB 4 is running... [3 time(s)]

JOB 4 is running... [4 time(s)]

JOB 2 is running... [9 time(s)]

JOB 4 is running... [5 time(s)]

Writable Smart Insert 16:1

Example 2:

Finish input of all four jobs in several quantum.

JOB 1 runs more than one quantum until JOB 2 arrives; JOB 1 and 2 run alternately until JOB 3 arrivals; JOB 1, 2, and 3 run alternately until one of them is completed, then JOB 4 begins to run.

The screenshot shows the Eclipse IDE with the following components:

- Package Explorer:** Shows the project structure with packages like LeetCode, Scheduling System, jobs, main, roundrobin, threads, and JRE System Library.
- Main.java:** Contains the following code:

```
1 package main;
2
3 import threads.*;
4
5 public class Main {
6     public static void main(String[] args) {
7         // Multithreading "CPU" & "Task" run
8         CPU.getInstance().start();
9
10        new Timer();
11
12        new ReadKeyboard();
13    }
14 }
15
16
17
18
```
- Read Keyboard Input Dialog:** A modal dialog with the title "Read Keyboard Input". It contains a table with columns "JOB No.", "Initial Value", and "End Value". The data is as follows:

JOB No.	Initial Value	End Value
1	1	40
2	1	20
3	1	12
4	1	8
- Console:** Shows the output of the program, including the message "The CPU scheduler is working..." and a series of status messages for each job, such as "JOB 1 is running... [1 time(s)]", "JOB 2 is running... [1 time(s)]", etc.

Java - Scheduling System/src/main/Main.java - Eclipse - /Users/Suming/workspace

Package Explorer: LeetCode, Scheduling System, src, jobs, AbstractJob.java, CounterJob.java, main, Main.java, roundrobin, Scheduler.java, threads, CPU.java, ReadKeyboard.java, Task.java, Timer.java, JRE System Library

Main.java:

```
1 package main;
2
3 import threads.*;
4
5 public class Main {
6     public static void main(String[] args) {
7         // Multithreading "CPU" & "Task" run
8         CPU.getInstance().start();
9
10        new Timer();
11
12        new ReadKeyboard();
13    }
14 }
15
16
17
18
```

Read Keyboard Input:

JOB No.	Initial Value	End Value
1	1	40
2	1	20
3	1	12
4	1	8

Console:

```
Main [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_45.jdk/
JOB 3 is running... [ 1 time(s) ]
JOB 3 is running... [ 2 time(s) ]
JOB 3 is running... [ 3 time(s) ]
JOB 3 is running... [ 4 time(s) ]
JOB 1 is running... [ 13 time(s) ]
JOB 1 is running... [ 14 time(s) ]
JOB 1 is running... [ 15 time(s) ]
JOB 1 is running... [ 16 time(s) ]
JOB 2 is running... [ 9 time(s) ]
JOB 2 is running... [ 10 time(s) ]
JOB 2 is running... [ 11 time(s) ]
JOB 2 is running... [ 12 time(s) ]
JOB 3 is running... [ 5 time(s) ]
JOB 3 is running... [ 6 time(s) ]
JOB 3 is running... [ 7 time(s) ]
JOB 3 is running... [ 8 time(s) ]
JOB 1 is running... [ 17 time(s) ]
JOB 1 is running... [ 18 time(s) ]
JOB 1 is running... [ 19 time(s) ]
JOB 1 is running... [ 20 time(s) ]
JOB 2 is running... [ 13 time(s) ]
JOB 2 is running... [ 14 time(s) ]
JOB 2 is running... [ 15 time(s) ]
```

Java - Scheduling System/src/main/Main.java - Eclipse - /Users/Suming/workspace

Package Explorer: LeetCode, Scheduling System, src, jobs, AbstractJob.java, CounterJob.java, main, Main.java, roundrobin, Scheduler.java, threads, CPU.java, ReadKeyboard.java, Task.java, Timer.java, JRE System Library

Main.java:

```
1 package main;
2
3 import threads.*;
4
5 public class Main {
6     public static void main(String[] args) {
7         // Multithreading "CPU" & "Task" run
8         CPU.getInstance().start();
9
10        new Timer();
11
12        new ReadKeyboard();
13    }
14 }
15
16
17
18
```

Read Keyboard Input:

JOB No.	Initial Value	End Value
1	1	40
2	1	20
3	1	12
4	1	8

Console:

```
Main [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_45.jdk/
JOB 2 is running... [ 13 time(s) ]
JOB 2 is running... [ 14 time(s) ]
JOB 2 is running... [ 15 time(s) ]
JOB 2 is running... [ 16 time(s) ]
JOB 3 is running... [ 9 time(s) ]
JOB 3 is running... [ 10 time(s) ]
JOB 3 is running... [ 11 time(s) ]
JOB 3 is running... [ 12 time(s) ]
JOB 4 is running... [ 1 time(s) ]
JOB 4 is running... [ 2 time(s) ]
JOB 4 is running... [ 3 time(s) ]
JOB 4 is running... [ 4 time(s) ]
JOB 1 is running... [ 21 time(s) ]
JOB 1 is running... [ 22 time(s) ]
JOB 1 is running... [ 23 time(s) ]
JOB 1 is running... [ 24 time(s) ]
JOB 2 is running... [ 17 time(s) ]
JOB 2 is running... [ 18 time(s) ]
JOB 2 is running... [ 19 time(s) ]
JOB 2 is running... [ 20 time(s) ]
JOB 4 is running... [ 5 time(s) ]
JOB 4 is running... [ 6 time(s) ]
JOB 4 is running... [ 7 time(s) ]
```

Java - Scheduling System/src/main/Main.java - Eclipse - /Users/Suming/workspace

Package Explorer: LeetCode, Scheduling System, src, jobs, AbstractJob.java, CounterJob.java, main, Main.java, roundrobin, Scheduler.java, threads, CPU.java, ReadKeyboard.java, Task.java, Timer.java, JRE System Library

Main.java

```
1 package main;
2
3 import threads.*;
4
5 public class Main {
6     public static void main(String[] args) {
7         // Multithreading "CPU" & "Task" run
8         CPU.getInstance().start();
9
10        new Timer();
11
12        new ReadKeyboard();
13
14    }
15 }
16
17
18
```

Read Keyboard Input

JOB No.	Initial Value	End Value
1	1	40
2	1	20
3	1	12
4	1	8

Console

Main [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_45.jdk/...

JOB 2 is running... [19 time(s)]

JOB 2 is running... [20 time(s)]

JOB 4 is running... [5 time(s)]

JOB 4 is running... [6 time(s)]

JOB 4 is running... [7 time(s)]

JOB 4 is running... [8 time(s)]

JOB 1 is running... [25 time(s)]

JOB 1 is running... [26 time(s)]

JOB 1 is running... [27 time(s)]

JOB 1 is running... [28 time(s)]

JOB 1 is running... [29 time(s)]

JOB 1 is running... [30 time(s)]

JOB 1 is running... [31 time(s)]

JOB 1 is running... [32 time(s)]

JOB 1 is running... [33 time(s)]

JOB 1 is running... [34 time(s)]

JOB 1 is running... [35 time(s)]

JOB 1 is running... [36 time(s)]

JOB 1 is running... [37 time(s)]

JOB 1 is running... [38 time(s)]

JOB 1 is running... [39 time(s)]

JOB 1 is running... [40 time(s)]