

Exercise Sheet 8

Aufgabe 1 (Interrupts)

1. What are interrupts?
2. What is the interrupt vector?
3. What are exceptions?
4. What happens, if during the handling of an interrupt, an additional interrupt occurs?

Aufgabe 2 (Scheduling Strategies)

1. Why exists a system idle process in some operating systems?
2. Explain the difference between preemptive and non-preemptive scheduling.
3. Name one drawback of preemptive scheduling.
4. Name one drawback of non-preemptive scheduling.
5. How does static multilevel scheduling work?
6. How does multilevel feedback scheduling work?
7. Which scheduling strategies are fair?

A scheduling method is „fair“ when each process gets the CPU assigned at some point.

- | | |
|--|--|
| <input type="checkbox"/> Priority-driven scheduling | <input type="checkbox"/> Shortest Remaining Time First |
| <input type="checkbox"/> First Come First Served | <input type="checkbox"/> Longest Remaining Time First |
| <input type="checkbox"/> Last Come First Served | <input type="checkbox"/> Highest Response Ratio Next |
| <input type="checkbox"/> Round Robin with time quantum | <input type="checkbox"/> Earliest Deadline First |
| <input type="checkbox"/> Shortest Job First | <input type="checkbox"/> Fair share |
| <input type="checkbox"/> Longest Job First | |

8. Which scheduling strategies operate preemptive?

- | | |
|--|---|
| <input type="checkbox"/> First Come First Served | <input type="checkbox"/> Longest Remaining Time First |
| <input type="checkbox"/> Round Robin with time quantum | <input type="checkbox"/> Fair share |
| <input type="checkbox"/> Shortest Job First | <input type="checkbox"/> Static multilevel scheduling |
| <input type="checkbox"/> Longest Job First | <input type="checkbox"/> Multilevel feedback scheduling |
| <input type="checkbox"/> Shortest Remaining Time First | |

9. For which scheduling strategies must the CPU runtime (= execution time) be known?

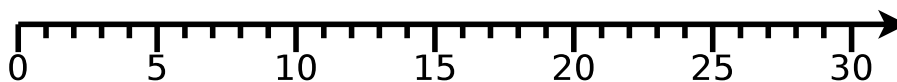
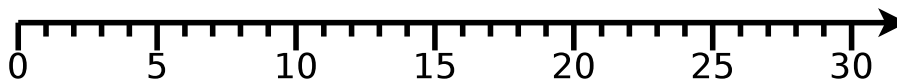
- | | |
|--|--|
| <input type="checkbox"/> Priority-driven scheduling | <input type="checkbox"/> Shortest Remaining Time First |
| <input type="checkbox"/> First Come First Served | <input type="checkbox"/> Longest Remaining Time First |
| <input type="checkbox"/> Last Come First Served | <input type="checkbox"/> Highest Response Ratio Next |
| <input type="checkbox"/> Round Robin with time quantum | <input type="checkbox"/> Earliest Deadline First |
| <input type="checkbox"/> Shortest Job First | <input type="checkbox"/> Fair share |
| <input type="checkbox"/> Longest Job First | |

Aufgabe 3 (Scheduling)

Process	CPU runtime [ms]	Creation time [ms]
A	5	0
B	7	3
C	2	5
D	6	7
E	1	10
F	5	18
G	4	24

1. Seven processes with different creation times shall be executed on a single CPU system. Draw the execution order of the processes with a Gantt chart (timeline) for...

- Longest Remaining Time First and
- Shortest Remaining Time First.

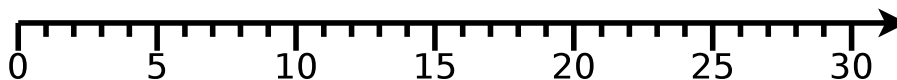
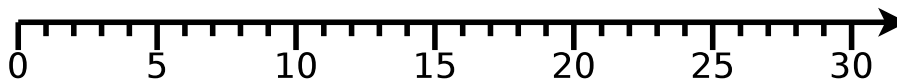
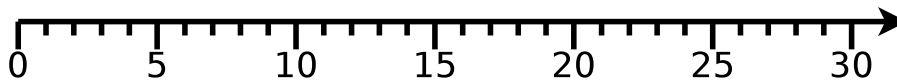
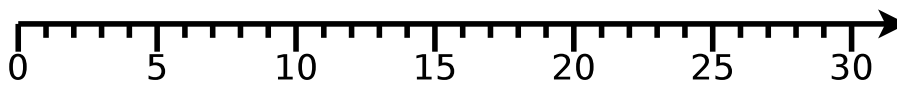
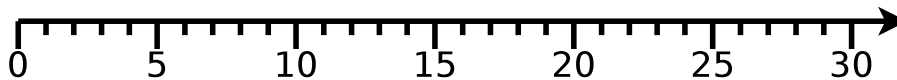


2. Calculate the average runtimes of the processes.
3. Calculate the average waiting times of the processes.

Aufgabe 4 (Scheduling)

Process	CPU runtime [ms]	Creation time [ms]
A	10	0
B	8	4
C	2	6
D	5	11
E	4	13
F	1	15

1. Six processes with different creation times shall be executed on a single CPU system. Draw the execution order of the processes with a Gantt chart (time-line) for...
 - First Come First Served,
 - Shortest Job First,
 - Longest Job First,
 - Shortest Remaining Time First and
 - Longest Remaining Time First.



2. Calculate the average runtimes of the processes.
3. Calculate the average waiting times of the processes.

Aufgabe 5 (Shell Scripts)

1. Program a shell script, which requests the user to select one of the four basic arithmetic operations. After selecting a basic arithmetic operation, the user is requested to enter two operands. Both operands are combined with each other via the previously selected basic arithmetic operation and the result is printed out in the following form:

<Operand1> <Operator> <Operand2> = <Result>

2. Modify the shell script from subtask 1 in a way that for each basic arithmetic operation a separate function exists. These functions should be relocated into an external function library and used for the calculations.
3. Program a shell script, which prints out a certain number of random numbers up to a certain maximum value. After starting the shell script, it should interactively query the values of these parameters:
 - Maximum value, which must be in the number range from 10 to 32767.
 - Desired number of random numbers.
4. Program a shell script, which creates the following empty files:

image0000.jpg, image0001.jpg, image0002.jpg, ..., image9999.jpg

5. Program a shell script, which renames the files from subtask 4 according to this scheme:

```
BTS_Exercise_<YEAR>_<MONTH>_<DAY>_0000.jpg
BTS_Exercise_<YEAR>_<MONTH>_<DAY>_0001.jpg
BTS_Exercise_<YEAR>_<MONTH>_<DAY>_0002.jpg
...
BTS_Exercise_<YEAR>_<MONTH>_<DAY>_9999.jpg
```