El-Oued University - Faculty of Exact Sciences
Department of Mathematics and Computer science
First-year Master in Internet of Things & Cybersecurity

Real Time Systems **2023 - 2024**

Exam correction

Exercise 01: 5

- We apply the acceptability test, in order to calculate the maximum execution time x of T5:

$$\sum_{i=1}^{n} \frac{C_i}{P_i} \leq n\left(2^{1/n} - 1\right)$$ 2

$$\frac{1}{10} + \frac{18}{100} + \frac{2}{20} + \frac{5}{50} + \frac{x}{25} \leq 0.743$$ 1

- We deduce that:

$$x \leq 6.575$$ 2

Exercise 02: 5

- LST algorithm:

	Time	Task	Slack time
0.5	0	T1	(4 - 0) – 2 = 2
		T2	(10 – 0) – 5 = 5
0.5	1	T1	(4 - 1) – 1 = 2
		T2	(10 – 1) – 5 = 4
0.5	2	T1	T1 has finished its execution.
		T2	(10 – 2) – 5 = 3
0.5	3	T1	T1 has finished its execution.
		T2	(10 – 3) – 4 = 3
0.5	4	T1	(4 - 0) – 2 = 2
		T2	(10 – 4) – 3 = 3
0.5	5	T1	(4 - 1) – 1 = 2

		T2	(10 – 5) – 3 = 2
0.5	6	T1	T1 has finished its execution.
		T2	(10 – 6) – 3 = 1
0.5	7	T1	T1 has finished its execution.
		T2	(10 – 7) – 2 = 1
0.5	8	T1	(4 - 0) – 2 = 2
		T2	(10 – 8) – 1 = 1
0.5	9	T1	(4 - 1) – 2 = 1
		T2	T2 has finished its execution.

Exercise 03: 10

```
Semaphore people=10;
Semaphore car=1;
Semaphore mutexpeople=1;
Semaphore mutexprio=1; // for the priority of people over cars
int nbpeople=0; // necessary to be able to wake up a car if the number of people
becomes 0
```

2

```
Process People ()
{
 p(people);// if there are already 10 people, block yourself
 p(mutexpeople);// protects the nbInd variable
 nbpeople++;
 if(nbpeople==1) // the first person arrives: they may or may not find a car
  {
   v(mutexpeople);
   p(car);// block cars (if there are no cars yet) or block himself
  }else v(mutexpeople);
 Cross_bridge();
 p(mutexpeople);
 nbpeople--;
 if(nbpeople==0) v(car);
 v(mutexpeople);
 v(people);
}
```

4

```
Process Cars ()
{
 p(mutexprio); // priority of people: cars cross two barriers (2 semaphores)
               // in such a way as to have a single car blocked at p(car).
 p(car);
 Cross_bridge();
 v(car);
 v(mutexprio);
}
```

4