Processes and Scheduling

The due dates for testation will be announced in Moodle!

Exercise 1: How to monitor processes and thread 

You are now familiar with top. Please try the following commands:

1. `ps -u $(whoami) -o ppid,pid,psr,sgi_p,pcpu,comm,policy,rtprio,pri,nice,time,c,f, wchan,cmd,pmem,maj_flt,min_flt,sz`

2. `pstree -acghlpsUu`

3. Combination with **`grep`** **e.g.** `ps -ef|grep $(whoami)`

Please try out the commands. Try to understand the results and what it means. Refer the man pages and/or google about the results. There is nothing to upload but you have to explain the results to the instructor.

Exercise 2: (prepare in the lab) 

Advice:

1. You can start a program directly with a specified RT-scheduling policy and a specified realtime priority on a specified CPU by using chrt in the following way:
   ```
   sudo /usr/bin/chrt -r 60 taskset 2 ./task $param1 $param2 … &
   ```

2. You can call command line commands from a C-program like this:
   ```
   char mycmd[100]="taskset 4 ./job ";
   system(mycmd);
   ```

3. You can concatenate strings by using strcat of <string.h> like
   ```
   char mycmd[100]="taskset 4 ./job ";
   strcat(mycmd, "param1");
   // leads to mycmd = "taskset 4 ./job param1"
   ```

4. A sub process inherits the scheduling policy and the priority of the super-process

Use your program/shell-script from exercise sheet IV: Modify it as the following way:

1. You have a shell script called taskgen.sh.
   This starts three programs task1, task2 and task 3.

2. You have three programs task1, task2 and task3.
   Each program **`task#`** starts periodically a program **`job`** to generate load.
   (this is a program like in Exercise-Sheet VI)
   The pseudo-code of program **`task#`** is as follows:

```
print $label, absolute_time
for i = 1 to 10
  start job $label $innerloop $no_of_time_slices
  nanosleep($delta_t_period)
  print $label, absolute_time
next i
```

with the following parameters:

`$label $innerloop $no_of_timeslices $delta_t_period`, e.g.:

```
./task  A  5200000  5  200000000  &
```

3. You have a program **`job`**:
   That just produces load for a specified execution time.

Explanation:

Job generates the load for a **`no_of_time_slices`** .

Task generates 10 Jobs with **`no_of_time_slices`** periodically with a specified period in nanoseconds.

With this setup you can run a scheduling very similar like RMS.

Make sure that all time stamps look similar and display the information you need (e.g. the label). Make sure that the attributes are separated by blanks or commas. When your program works you should pipe the output into an output file. Load this output file with Libre Office Calc. Sort the date by the column that contains the absolute time. After this you can easily analyze how the scheduler has scheduled you Jobs.

Exercise 3: (prepare at home) 

Please use the scheduling template provide in Moodle.

Rate Monotonic Scheduling (RMS)

1. A System has 4 periodic tasks (k=1, .., 4):
   $\mathbb{T} = \{ T^k \} = \{ (\Delta t_{exec}^k , \Delta t_{per}^k) \}$ with
   $T_1(7; 24)$; $T_2(1; 8)$; $T_3(4; 14)$ and $T_4(3; 15)$.
   Construct a schedule using RMS in the interval [0,50]!

2. A System has 3 periodic tasks $T^k = (\Delta t_{exec}^k , \Delta t_{per}^k)$ with
   a) $\mathbb{T} = \{ T^1(6; 20); T^2(3; 12); T^3(4; 10) \}$
   Construct a schedule using RMS in the interval [0,50]!

   b) $\mathbb{T} = \{ T^1(7; 20); T^2(3; 12); T^3(4; 10) \}$
   Construct a schedule using RMS in the interval [0,50]!

3. Calculate the load of the examples in 1. and 2. .

4. Calculate the schedulability test for the examples in 1. and 2.

Earliest Deadline First (EDF)

5. A System has 4 periodic tasks $T^k = (\Delta t_{exec}^k , \Delta t_{per}^k)$ with
   $\mathbb{T} = \{ T^1(7; 24); T^2(1; 8); T^3(4; 14); T^4(3; 15) \}$.
   Construct a schedule using EDF for the System.

6. A System has 3 periodic tasks $T^k = (\Delta t_{exec}^k , \Delta t_{per}^k)$ with
   a) $\mathbb{T} = \{ T^1(6: 20); T^2 (3; 12); T^3(4; 10) \}$
   Construct a schedule using EDF for the System in the interval [0,50]!

   b) $\mathbb{T} = \{ T_1(7; 20); T_2(3; 12); T_3(4; 10) \}$
   Construct a schedule using EDF for the System in the interval [0,50]!