

# Large Scale Computing

## Lab 2

Oscar Teeninga

1. Proszę wysłać zadanie wypisujące "Hello world"

```
#!/bin/bash
#SBATCH -p plgrid-testing
#SBATCH --ntasks-per-node=1
echo "Hello World!"
```

```
[[zeus][plgteeninga@login01 lab2]$ sbatch -N 1 -p plgrid-testing hello-world.sh
Submitted batch job 9866802
```

```
[[zeus][plgteeninga@login01 lab2]$ cat slurm-9866802.out
Hello World!
[zeus][plgteeninga@login01 lab2]$
```

2. Proszę wysłać zadanie, które wykonuje się dłużej, niż zadeklarowany czas (używać np.sleep) i sprawdzić, czy zadanie jest wstrzymywane

```
#!/bin/bash
#SBATCH -p plgrid-testing
#SBATCH --ntasks-per-node=1
sleep 120
echo "Udało sie :)"
```

```
[[zeus][plgteeninga@login01 lab2]$ sbatch -N 1 -p plgrid-testing -t 00:00:10 sleep.sh
Submitted batch job 9866819
```

```
slurmstepd: error: *** JOB 9866819 ON n1088-amd CANCELLED AT 2021-10-19T08:43:17 DUE TO TIME LIMIT ***
```

3. Proszę wysłać serię zadań używając zadań parametrycznych, z których każde wypisze jedną linię z pliku tekstowego. Należy użyć poleceń cat, head i tail.

```
#!/bin/bash
#SBATCH -N 1
#SBATCH -p plgrid-testing
#SBATCH --array=1-3
#SBATCH --ntasks-per-node=3
if [ $SLURM_ARRAY_TASK_ID = 1 ]; then
    cat text.txt | head -n 1
elif [ $SLURM_ARRAY_TASK_ID = 2 ]; then
    cat text.txt | head -n 2 | tail -n 1
else
    cat text.txt | tail -n 1
fi
```

```
[[zeus][plgteeninga@login01 lab2]$ cat text.txt
To jest linijka 1
To jest linijka 2
To jest linijka 3
```

```
[[zeus][plgteeninga@login01 lab2]$ sbatch printer.sh
Submitted batch job 9866934
[[zeus][plgteeninga@login01 lab2]$ ls
hello-world.sh  printer.sh  sleep.sh  slurm-9866934_1.out  slurm-9866934_2.out  slurm-9866934_3.out  text.txt
[[zeus][plgteeninga@login01 lab2]$ cat slurm-9866934_1.out
To jest linijka 1
[[zeus][plgteeninga@login01 lab2]$ cat slurm-9866934_2.out
To jest linijka 2
[[zeus][plgteeninga@login01 lab2]$ cat slurm-9866934_3.out
To jest linijka 3
```

4. Użyć zadań parametrycznych do renderowania serii klatek animacji przy użyciu programu povray.

```
#!/bin/bash
#SBATCH -p plgrid-testing
#SBATCH -N 1
#SBATCH --ntasks-per-node=10
#SBATCH --array=1-10
povray Subset_Start_Frame=$SLURM_ARRAY_TASK_ID Subset_End_Frame=$((SLURM_ARRAY_TASK_ID+1)) animation_a_.ini
```

```
[[zeus][plgteeninga@login01 animacja]$ ls -a
.          animation_a_.pov      animation_a_04.png  animation_a_08.png  slurm-9867254_10.out  slurm-9867254_8.out
..         animation_a_01.png    animation_a_05.png  animation_a_09.png  slurm-9867254_5.out  slurm-9867254_9.out
animation.sh animation_a_02.png    animation_a_06.png  animation_a_10.png  slurm-9867254_6.out
animation_a_.ini animation_a_03.png    animation_a_07.png  animation_a_11.png  slurm-9867254_7.out
```

```
[[zeus][plgteeninga@login01 animacja]$ cat animation_a_.ini
Antialias=On
Antialias_Threshold=0.3
Antialias_Depth=3
Width=12800
Height=7200
Input_File_Name=animation_a_.pov

Initial_Frame=1
Final_Frame=10
Initial_Clock=0
Final_Clock=1

Cyclic_Animation=on
Pause_when_Done=off
```

#### Frame Processing Times

```
Parse Time:    0 hours  0 minutes  0 seconds (0 seconds)
Photon Time:   0 hours  0 minutes  0 seconds (0 seconds)
Render Time:   0 hours  1 minutes 17 seconds (77 seconds)
Total Time:    0 hours  1 minutes 17 seconds (77 seconds)
```

#### Render Statistics

Image Resolution 12800 x 7200

```
-----
Pixels:        184345600  Samples:        184345600  SmpIs/Pxl: 1.00
Rays:          184345600  Saved:          0        Max Level: 1/5
-----
```

Zwiększyłem bazową rozdzielczość 10x w każdym wymiarze (liczba pikseli wzrosła 100x). Aby skrypt zadziałał trzeba było skorzystać z biblioteki povray w wersji 3.6 (wersja 3.7.0 była uszkodzona). Niestety wygenerowane obrazy są czarne, zapewne ze względu na wersję, natomiast nie otrzymałem nigdzie żadnego błędu.