



# Ejecución de arrays

La ejecución de arrays requerirá de una preparación previa por parte del usuario.

El usuario deberá estructurar los datos de entrada para las ejecuciones en carpetas o nombres de ficheros identificados mediante un número de trabajo, de manera que éste se pueda manipular de manera sencilla dentro del script de lanzamiento.

Una vez se tengan los datos de entrada organizados de esta manera, se puede proceder a la ejecución de los trabajos, lanzando en un script de submit tantas tareas como se necesiten.

## Ejecución de un array de 10 trabajos a un nodo

```
#!/bin/bash

#SBATCH -J <job_name>
#SBATCH -p <partition>
#SBATCH -N 1
#SBATCH -N <nodes>
#SBATCH --task=<number>
#SBATCH --cpus-per-task=<number>
#SBATCH --constrains=<node architecture> # sandy, ilk (icelake)... architecture
#SBATCH -t <days-HH:MM:SS>
#SBATCH --array=0-10
## %A JobId de la tarea principal (script de lanzamiento del arrayjob)
## %a Índice de la tarea dentro del arrayjob
## %j JobId de la tarea dentro del arrayjob en el gestor de colas
#SBATCH -o <array-%A-%a-%j.out>
#SBATCH -D .
#SBATCH --mail-type END
#SBATCH --mail-user <mail to notify>
#####

module purge
module load <modules>

echo "Array job $SLURM_ARRAY_JOB_ID -> $SLURM_ARRAY_TASK_ID"
srun program
echo "End array job $SLURM_ARRAY_JOB_ID -> $SLURM_ARRAY_TASK_ID"
```

## Ejecución de un array de 100 trabajos con sólo 10 trabajos simultáneos

```
#!/bin/bash

#SBATCH -J <job_name>
#SBATCH -p <partition>
#SBATCH -N 1
#SBATCH -t <days-HH:MM:SS>
#SBATCH -array=0-100%10
## %A JobId de la tarea principal (script de lanzamiento del arrayjob)
## %a Índice de la tarea dentro del arrayjob
## %j JobId de la tarea dentro del arrayjob en el gestor de colas
#SBATCH -o <array-%A-%a-%j.out>
#SBATCH -D .
#SBATCH --mail-type END
#SBATCH --mail-user <mail to notify>
#####

module purge
module load <modules>

echo "Array job $SLURM_ARRAY_JOB_ID -> $SLURM_ARRAY_TASK_ID"
srun program
echo "End array job $SLURM_ARRAY_JOB_ID -> $SLURM_ARRAY_TASK_ID"
```