Example scripts: peak_memusage

Return to [Checking memory use](#).

Page contents

Serial and OpenMP jobs with peak_memusage

Be sure to substitute your own project code, job name, executable name, wall-clock time (hours:minutes:seconds) and so on when customizing a sample script to run your job. Specify input or output redirection as you normally do if needed. For parallel jobs, you also will likely need to adjust the node count and possibly tasks per node.

```
#!/bin/tcsh
#PBS -N peakmem
#PBS -A project_code
#PBS -l walltime=01:00:00
#PBS -q share
#PBS -j oe
#PBS -k eod
#PBS -m abe
#PBS -M email_address
#PBS -l select=1:ncpus=1

setenv TMPDIR /glade/scratch/$USER/temp
mkdir -p $TMPDIR

### Load the module environment
module purge
module load ncarenv intel
module load ncarcompilers
module load peak_memusage
module li

### Run the executable
peak_memusage.exe ./executable_name --arguments
```

The output will include a line like the following one.

```
Used memory in task 0/1: 381.99MiB (+0.67MiB overhead). ExitStatus: 0. Signal: 0
```

The "overhead" identified in the output is memory that the tool uses to check your program. If the program exits unsuccessfully or if it receives a signal, the exit status and signal number also will be printed.

MPI and hybrid jobs with peak_memusage

Be sure to substitute your own project code, job name, executable name, wall-clock time (hours:minutes:seconds) and so on when customizing a sample script to run your job. Specify input or output redirection as you normally do if needed. For parallel jobs, you also will likely need to adjust the node count and possibly tasks per node.

For an MPT-compiled binary

```
#!/bin/tcsh
#PBS -N peakmem
#PBS -A project_code
#PBS -l walltime=01:00:00
#PBS -q regular
#PBS -j oe
#PBS -k eod
#PBS -m abe
#PBS -M user_email_address
#PBS -l select=2:ncpus=36:mpiprocs=36

setenv TMPDIR /glade/scratch/$USER/temp
mkdir -p $TMPDIR

### Load the module environment
module purge
module load ncarenv intel
module load ncarcompilers mpt
module load peak_memusage
module li

# yyyy-mm-dd Context: Cheyenne MPT peak_memusage job.
# Variable MPI_SHEPHERD is set in this job in order to
# enable peak_memusage. Do not propagate it to other MPT
# jobs as it may cause significant slowdown or timeout.
# Contact the CISL Consulting Services Group if you have
# questions about this.
setenv MPI_SHEPHERD true

### Run the executable
mpiexec_mpt peak_memusage.exe ./executable_name --arguments
```

For an Intel MPI-compiled binary

```
#!/bin/tcsh
#PBS -N peakmem
#PBS -A project_code
#PBS -l walltime=01:00:00
#PBS -q regular
#PBS -j oe
#PBS -k eod
#PBS -m abe
#PBS -M email_address
#PBS -l select=2:ncpus=36:mpiprocs=36

setenv TMPDIR /glade/scratch/$USER/temp
mkdir -p $TMPDIR

### Load the module environment
module purge
module load ncarenv intel
module load ncarcompilers impi
module load peak_memusage
module li

### Run the executable
mpirun peak_memusage.exe ./executable_name --arguments
```

The output for either of the MPI jobs above will include a line for each MPI task that the program used, as in this fragment.

```
Used memory in task 69/36: 97.36MiB (+31.10MiB overhead). ExitStatus: 0. Signal: 0
Used memory in task 40/36: 173.65MiB (+31.10MiB overhead). ExitStatus: 0. Signal: 0
Used memory in task 43/36: 59.21MiB (+31.10MiB overhead). ExitStatus: 0. Signal: 0
Used memory in task 11/36: 137.97MiB (+31.09MiB overhead). ExitStatus: 0. Signal: 0
Used memory in task 1/36: 1.23MiB (+0.78MiB overhead). ExitStatus: 0. Signal: 0
```

Complete output would show used memory for tasks 0-71. The "overhead" identified in the output is memory that the tool uses to check your program. If the program exits unsuccessfully or if it receives a signal, the exit status and signal number also will be printed.