

北航高性能计算平台使用常见问题解答

1.0 版本

北京航空航天大学信息办高算中心

2020 年 12 月

目 录

一、	用户登录与连接相关	1
二、	环境安装与设置相关	3
三、	作业提交与调度相关	9
四、	应用计算相关	16

一、用户登录与连接相关

Q1: VPN 无法连接，或者连接超时。

A: 下载并重新安装 easy connect 下载地址：
<https://vpn.buaa.edu.cn/portal/#!/login>，输入正确 VPN 地址：
<https://vpn.buaa.edu.cn>

注：北航校内网环境下无需使用 VPN，可直连 HPC 平台。

Q2: VPN 账户密码遗忘。

A: 登录 VPN 使用的是统一认证账号密码，找回密码请登录
ucs.buaa.edu.cn，选择通过手机找回，输入学号/教工号即可。

Q3: HPC 平台账户密码遗忘。

A: 用户交流群中联系平台管理员协助进行密码重置，或者发送
问题说明邮件至 hpc@buaa.edu.cn。

Q4: SSH 连接到平台，长时间不操作，会话自动断开。

A: 不同的 ssh 工具都有默认会话保持设置，请检查此项设置，
若是频繁自动断开，请检查本地网络环境。

Q5: 在登录节点编译运行时出现终端自动断开现象。

A: 编译会占用登录节点 CPU 资源，若超过系统设定阈值，即：
进程 CPU 使用率 $\geq 100\%$ ，且运行时间超过 15 分钟、进程内存使用大
小超过 10GB，则会被系统监控查杀进程。cp、scp、tar 传输解压命
令占用超过阈值除外，建议将编译过程写成调度脚本，通过调度作业
编译。

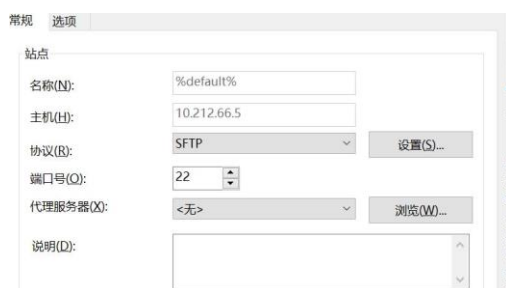
Q6: SSH 登录平台，输入错误的用户名密码后再输入正确的用户名密码却提示密码错误，拒绝连接。

A: 短时间内登录输错 3 次密码，那么登录时所使用 IP 将被自动封锁 5 分钟。可尝试换登录节点（10.212.66.4、10.212.66.5、10.212.70.128）登录，3 个节点数据共享，锁定机制独立，或者等待 5 分钟再认证。

Q7: 使用终端如 xftp 无法从平台下载文件。

状态	进度	大小
错误	2%	11.91MB/432.23MB
错误	2%	11.91MB/432.23MB
错误	31%	800KB/2.48MB
错误	4%	21.53MB/432.23MB
错误	1%	6.94MB/432.23MB

A: 建议使用 xshell 终端，使用 sz 文件名下载，若使用 XFTP，则检查如下参数。



Q8: 原来都是在 windows 上操作软件，不会用 Linux，我现在想学习一些基本的 Linux 操作，满足超算上机需要，应该学哪些内容？

A: 首先了解一下 Linux 的目录结构，Linux 的目录以树形组织。

其次学习几个常用的 Linux 命令，主要是：

pwd -查看当前所在目录

cd -改变目录

cp -复制文件

rm -删除文件

mv -文件重命名

mkdir -创建目录

然后学习作业管理相关的命令：

sbatch -提交作业

scancel job-取消作业

scontrol show job -查看作业状态

sinfo -查看节点信息

有关 slurm 的使用请参考北航校级计算平台用户手册或相关网络资料。

二、环境安装与设置相关

Q9： 是否可以提供 root 或者 sudo 权限，用来进行软件或者环境包安装

A： root 或者 sudo 权限，任何情况下，都不会提供给用户。

如果是需要执行 yum install 这样的操作，平台大部分依赖包已基本安装好，如果缺少相关的软件包，请把软件包名字告诉超算管理员，让管理员来安装。

如果是执行 apt install，需要注意北航 HPC 平台为 centos 7

系统，不是 Debian/Ubuntu 系统，需要安装的包可能在 centos 叫其他的名字，如果你后续的安装使用步骤提示确实缺少依赖包，请把这个依赖包的名字告知超算管理员，让管理员来安装。

Q10: 如何在主机上安装软件？

A: (a) 如果有开源源码包，可以直接在自己家目录编译安装；

(b) 使用 anaconda 环境安装，比如安装 opencv、pytorch、TensorFlow；

(c) 如果有正版软件许可，可以联系管理进行安装，例如：Comsol、ABAQUS；

注意：平台不建议用户安装破解或者盗版软件，用户须自行负责相关软件的版权问题。

Q11: 我想在主机上自行编译软件，需注意哪些问题？

A: (a) 首先确定您的软件的系统要求、编译器版本要求、依赖的软件包，主机上已预装了一部分工具软件，例如查看系统预装依赖

包：`rpm -qa | grep 包名称`

```
[paratera@ln02 ~]$ rpm -qa | grep -i opencv
opencv-2.4.5-3.el7.x86_64
opencv-devel-2.4.5-3.el7.x86_64
opencv-core-2.4.5-3.el7.x86_64
[paratera@ln02 ~]$ rpm -qa | grep -i fftw
fftw-3.3.3-8.el7.x86_64
fftw-libs-double-3.3.3-8.el7.x86_64
fftw-libs-long-3.3.3-8.el7.x86_64
fftw-libs-single-3.3.3-8.el7.x86_64
[paratera@ln02 ~]$
```

(b) 设置好编译环境，准备好软件包来进行编译测试；

```
-- Install configuration: ""  
CMake Error at cmake_install.cmake:36 (FILE):  
  file cannot create directory: /usr/local/doc/cmake-2.8.  Maybe need  
  administrative privileges.  
  
gmake: *** [install] Error 1
```

如上图所示，一般编译安装软件没有修改默认的安装路径(通常是 /opt /usr/local 这样需要 root 权限的系统目录)。要解决这类问题，如果是从源代码编译软件，一般是在 configure 的时候使用 --prefix= 这个选项把安装目录指定到用户自己的目录下。如果是安装型的软件，在安装向导里修改默认的安装目录为用户自己的家目录。

注意：如果是大型软件，编译测试可能需要占用大量 CPU 的，请使用调度系统编译测试。

(c) 如果安装包是 ISO 格式文件，请在本地解压 ISO 文件内容，并打包成 zip 格式压缩包再上传至平台安装。因为 ISO 文件只有 root 权限可以使用 mount 命令挂载。

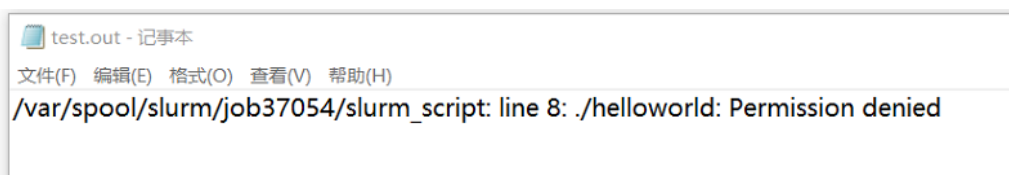
Q12: 发现主机的编译器版本很低，无法满足我的需要，怎么办？

A: 主机出于系统稳定性、可用性的需要，编译器和库文件的更新周期较长，因此建议您：

(a) 联系我方技术服务人员，或者自行在目录下安装编译器和其他工具软件，注意需在 .bashrc 里设置相应的路径；

(b) 一些系统底层的 api 如 glibc 的版本无法进行更新。

Q13: 作业执行时提示，程序无执行权限问题。如图



A: 建议使用 `chmod` 命令给执行文件加执行权限, `chmod +x` 即可。

Q14: 如何使用 `mpif9`、`mpicc`、`mpirun` 等命令?

A: 使用 `module load intel` 命令加载, 示例如下:

```
(base) [test@ln02 ~]$ module load intel
(base) [test@ln02 ~]$ which mpif90
/gs/software/intel/compilers_and_libraries_2019.5.281/linux/mpi/intel64/bin/mpif90
(base) [test@ln02 ~]$ which mpirun
/gs/software/intel/compilers_and_libraries_2019.5.281/linux/mpi/intel64/bin/mpirun
(base) [test@ln02 ~]$ which mpicc
/gs/software/intel/compilers_and_libraries_2019.5.281/linux/mpi/intel64/bin/mpicc
```

Q15: 如何查看 `gcc` 的路径以及相关信息?

A: 查看 `gcc5` 使用如下命令:

```
[paratera@ln02 ~]$ module load gnu
[paratera@ln02 ~]$ gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/gs/software/ohpc/
Target: x86_64-unknown-linux-gnu
Configured with: ../configure --disabl
Thread model: posix
gcc version 5.3.0 (GCC)
```

查看 `gcc8` 使用如下命令:

注意, 需要先 `module unload gnu`, 避免冲突。

```
[paratera@ln02 ~]$ module unload gnu
[paratera@ln02 ~]$ module load gnu8
[paratera@ln02 ~]$ gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/gs/software/ohpc/pub/compile
Target: x86_64-pc-linux-gnu
Configured with: ../configure --disable-multilib
Thread model: posix
gcc version 8.3.0 (GCC)
```

Q16: 如何使用 R, R 语言依赖于 `gnu8`, 所以需要先加载 `gnu8`,

如直接使用 `module load R`, 会提示找不到模块。

A: 使用如下命令:

```
[paratera@ln02 ~]$ module load gnu8
[paratera@ln02 ~]$ module load R
[paratera@ln02 ~]$ R --version
R version 3.5.3 (2019-03-11) -- "Great Truth"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under the terms of the
GNU General Public License versions 2 or 3.
For more information about these matters see
http://www.gnu.org/licenses/.

[paratera@ln02 ~]$
```

Q17: 如何使用 openmpi, Openmpi 依赖于 intel 的 mpi 模块,
所以需要先 module load intel 。

A: 使用如下命令:

```
[root@mgmt01 ~]# module load intel
[root@mgmt01 ~]# module load openmpi3
[root@mgmt01 ~]# which mpirun
/opt/ohpc/pub/mpi/openmpi3-intel/3.1.4/bin/mpirun
```

Q18: 如何删除乱码文件?

```
?b??FZr?????h???=?:?t$????@s???v6J?
??G????k??tdi?5???n?u??
G????k??tdi?5???n?u??
H4mm_Co_0_7.mat
H4mm_T_0_7.mat
```

A: 首先使用 ls 命令定位具体文件: ls -i 查找文件 i 节点名,
然后再使用 find 命令进行删除: find -inum 节点号 -delete 删除
节点号文件。如图:

```
[root@ln02 test]# ls -i
872630879 526gpu.txt      872625205 531cpu.txt      872625208 531userjob.txt
872630878 526usercpu.txt      872625206 531gpu.txt      872621652 userjob.txt
[root@ln02 test]# find -inum 872630879 -delete
[root@ln02 test]# ls -i
872630878 526usercpu.txt      872625206 531gpu.txt      872621652 userjob.txt
872625205 531cpu.txt      872625208 531userjob.txt
```

Q19: Q: 如何使用 anaconda 安装软件

A: 可以根据用户手册, module 与 anaconda 章节, 自定义环境配置与软件安装部分, 例如安装 Pytorch, 自定义配置安装 conda 环境,
并使用 pip install python=3.6 pip install pytorch 即可安装相

应软件。

Q20: 在自定义 conda 环境安装了某软件,提交作业后提示找不到软件模块,比如:

```
Traceback (most recent call last):
  File "./test.py", line 1, in <module>
    import torchvision
ModuleNotFoundError: No module named 'torchvision'
(chenenv) [chentianyou@ln01 execute]$
```

A: conda activate 自定义环境,并使用 conda list | grep 软件名,查看当前 conda 环境是否有此软件;若有,再执行 import 软件名,看是否有报错,并确定软件 python 版本是否匹配,若正常则修改脚本,脚本中可以添加以下命令:

```
module load anaconda3
source activate testenv (比如: TensorFlow 安装在了 testenv 环境中)
```

并退出到 base 环境(anaconda)提交作业,conda deactivate 即可退出。

Q21: 用户之间如何进行数据共享?

A: 平台中的目录/gs/sharedata/用户名,即为用户共享目录,不同用户可进行数据共享,共享完数据,请随即删除数据,释放存储空间。

注:共享空间也受单用户 2T 磁盘配额限制。

Q22: mac 和 Linux 操作系统与高算平台如何传输文件?

A: 无需下载传输工具,使用 MAC 或 LINUX 系统自带的 scp 命令即可实现。

1) 上传单个文件:

```
scp /users/test/Desktop/helloworld.c test@10.212.66.5:/gs/home/test
```

2) 下载单个文件:

```
scp test@10.212.66.5:/gs/home/test/helloworld.c /users/test/Desktop
```

3) 上传单个文件夹:

```
scp -r /users/test/Desktop/test test@10.212.66.5:/gs/home/test
```

4) 下载单个文件夹

```
scp -r test@10.212.66.5:/gs/home/test/test /users/test/Desktop
```

三、作业提交与调度相关

Q23: 提交脚本报错如下, DOS line、UNIX line 等。

```
[jiangzhenhua@ln01 11]$ sbatch 11.txt
sbatch: error: Batch script contains DOS line breaks (\r\n)
sbatch: error: instead of expected UNIX line breaks (\n).
```

A: 此问题多为 Windows 平台编写命令直接上传到平台导致, 使用 dos2unix 命令转换脚本, 建议使用 vim 命令编写脚本。

Q24: 提交的作业没有实时输出信息。

A: 可以尝试在 print 函数里面加上 ",flush=True" 参数, 让打印信息的时候刷新标准输出。

Q25: 作业提交后无法使用 nvcc、nvidia-smi 等需要调用 GPU 程序的命令, 或者找不到 libcuda.so 文件。

```
[lihao815@ln01 darknet]$ make
gcc -Iinclude/ -Isrc/ -DGPU -I/usr/local/cuda/include/ -Wall -Wno-unused-result -Wno-unknown-pragmas -Wfatal-errors -fPIC -Ofast -DGPU -c ./src/gemm.c -o obj/gemm.o
In file included from ./src/utils.h:5:0,
                 from ./src/gemm.c:2:
include/darknet.h:14:30: fatal error: cuda_runtime.h: No such file or directory
#include "cuda_runtime.h"
                           ^
compilation terminated.
make: *** [obj/gemm.o] Error 1
[lihao815@ln01 darknet]$ srun -p gpu --gres=gpu:1 bash
srun: job 65904 queued and waiting for resources
```

A: 检查脚本是否提交到了 GPU 分区、是否有添加 #SBATCH --

gres=gpu:1 参数。如果使用的是 srun 模式，需要先申请 GPU 卡资源，再进行程序交互测试。

Q26: 申请到 gpu 卡，输入 `nvcc -V` 提示找不到命令。

A: 使用命令 `module load cuda/10.1` 加载模块，或者个人环境变量 `.bashrc` 中加入：

```
export PATH=/usr/local/cuda/bin:$PATH
```

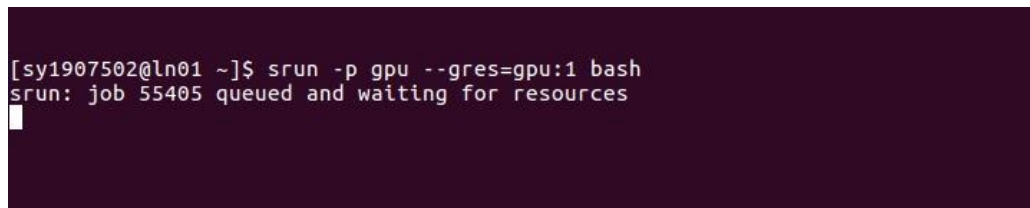
Q27: 提交作业已成功运行，但是运行结果提示 killed。

A: 此类问题多为计算程序在节点使用了过多的内存，被系统杀死。建议修改计算程序的模块尺寸等与内存相关的参数，或者脚本中添加 `--mem=<size[units]>` 参数，具体使用可参考 `sbatch` 命令 `man` 手册：`man sbatch | grep -i mem`

Q28: 提交的作业占用内存较大，会挤掉其他用户的作业，报错：`out of memory`

A: 通常这种情况多数出现在 gpu 节点，计算如 TensorFlow 的程序，此类情况建议用户可以降低程序的图复杂度，或者降低 `batch size`；如果是图片的话可以将 `resize` 调整小一些。

Q29: 任务或者作业已经成功提交，显示在排队，如何知道需要排队多久。



```
[sy1907502@ln01 ~]$ srun -p gpu --gres=gpu:1 bash
srun: job 55405 queued and waiting for resources
```

A: 等待时间根据平台当前资源而定，具体可以使用命令 `scontrol`

show job jobid 查看如下信息。

```
SubmitTime=2020-05-30T19:51:05 EligibleTime=2020-05-30T19:51:05
AccrueTime=2020-05-30T19:51:05
StartTime=2020-06-20T10:21:54 EndTime=2020-06-23T10:21:54 Deadline=N/A
SuspendTime=None SecsPreSuspend=0 LastSchedEval=2020-06-19T13:33:29
```

Q30: 提交的任务已经排队很久了，依旧没有正常运行。

```
JOBID PARTITION NAME USER ST TIME NODES NODELIST(REASON)
203808 normal CuPd jma008 PD 0:00 1 (PartitionTimeLimit)
203809 normal CuPd jma008 PD 0:00 1 (PartitionTimeLimit)
203810 normal CuPd jma008 PD 0:00 1 (PartitionTimeLimit)
203811 normal CuPd jma008 PD 0:00 1 (PartitionTimeLimit)
203812 normal CuPd jma008 PD 0:00 1 (PartitionTimeLimit)
203813 normal CuPd jma008 PD 0:00 1 (PartitionTimeLimit)
203814 normal CuPd jma008 PD 0:00 1 (PartitionTimeLimit)
```

A: 使用 `squeue -u 用户名` 查看作业目前状态，根据状态来定位原因，如上图 `PartitionTimeLimit`，即超过分区时间最大 7 天的限制，此类作业会一直排队，无法执行，请遵循平台用户作业限制规范重新提交。

Q31: `srun` 和 `sbatch` 提交任务有什么区别

A: `srun` 是调试模式，是交互模式操作，用户程序调试，如果网络问题，或者人为断开终端，则申请资源即刻释放。

`sbatch` 类似后台命令提交，作业提交成功后，若程序没问题，申请资源不受终端断开，或者网络原因而中断。

Q32: 提交的作业报错信息为超时，如果修改执行时间，任务是否会继续执行。

A: 任务会终止，资源随即释放，需要重新提交，建议任务时间不超过系统规定的 7 天时间。

Q33: 如何在系统中对程序进行调试？

A: 对程序的调试可以通过作业调度系统分配到计算节点上。`srun`

和 `salloc` 都可以用于交互式调试应用，当调试的计算量比较大时，请不要在登录节点上直接运行，高负载的用户进程会被系统守护进程杀掉。

具体操作如下（以分配 GPU 卡为例）：

1) 执行：

```
srun -n 1 -p gpu -gres=gpu:1 /bin/bash
```

接着执行：

```
nvidia-smi
```

可以看到系统分配了一块 GPU 卡。中断交互调试，任务结束。

2) 执行：

```
sbatch -n 1 -p gpu -gres=gpu:1 /bin/bash
```

接着执行：

```
srun nvidia-smi
```

可以看到系统分配了一块 GPU 卡。

退出终端，任务结束。

Q34: 如何获得作业运行时环境变量？

A: Slurm 作业调度系统运行时输出的主要环境变量如下：

SLURM_ARRAY_TASK_ID: 作业组 ID（索引）号；

SLURM_ARRAY_TASK_MAX: 作业组最大 ID 号；

SLURM_ARRAY_TASK_MIN: 作业组最小 ID 号；

SLURM_ARRAY_TASK_STEP: 作业组索引步进间隔;

SLURM_ARRAY_JOB_ID: 作业组主作业号;

SLURM_CLUSTER_NAME: 集群名;

SLURM_CPUS_ON_NODE: 分配的节点上的 CPU 数量;

SLURM_CPUS_PER_TASK: 每个任务的 CPU 数量;

SLURM_JOB_ID 作业号;

SLURM_JOB_CPUS_PER_NODE 每个节点上的 CPU 颗数;

SLURM_JOB_DEPENDENCY: 作业依赖信息, 由 `-dependency` 选项设置;

SLURM_JOB_NAME: 作业名;

SLURM_JOB_NODELIST: 分配的节点名列表;

SLURM_JOB_NUM_NODES: 分配的节点总数;

SLURM_JOB_PARTITION: 使用的队列名;

SLURM_JOB_RESERVATION: 作业预留;

SLURM_LOCALID: 节点本地任务号;

SLURM_MEM_PER_CPU: 类似 `-mempercpu`, 每颗 CPU 的内存

SLURM_MEM_PER_NODE: 类似 `-mem`, 每个节点的内存;

SLURM_NODE_ALIASES: 分配的节点名、通信 IP 地址和主机名组合;

SLURM_NODEID: 分配的节点号;

SLURM_NTASKS: 类似 `-n`, `-ntasks`, 总任务数, CPU 核数;

SLURM_NTASKS_PER_CORE: 每个 CPU 核分配的任务数;

SLURM_NTASKS_PER_NODE: 每个节点上的任务数;

SLURM_NTASKS_PER_SOCKET: 每颗 CPU 上的任务数, 仅 -ntaskspersocket 选项设定时设定;

SLURM_PRIO_PROCESS: 进程的调度优先级 (nice 值);

SLURM_PROCID: 当前进程的 MPI 秩;

SLURM_PROFILE: 类似 -profile;

SLURM_RESTART_COUNT: 因为系统失效等导致的重启次数;

SLURM_SUBMIT_DIR: sbatch 启动目录, 即提交作业时目录;

SLURM_SUBMIT_HOST: sbatch 启动的节点名, 即提交作业时节点;

SLURM_TASKS_PER_NODE: 每节点上的任务数;

SLURM_TASK_PID: 任务的进程号 PID; SLURMD_NODENAME: 执行作业脚本的节点名。

Q35: 作业执行不正常时如何查看原因

A: 用户提交作业后, 是否运行取决于用户申请的资源情况和当前系统的情况。建议使用 `squeue` 命令来查看所有已经提交和正在运行的作业。其中 `NODELIST(REASON)` 一栏包含非常有用的信息, 在作业未运行时, 它会显示未运行的原因; 当作业在运行时, 它会显示作业是在哪个节点运行的。

下面是作业未正常运行的原因:

原因代码	详细说明
BeginTime	未到用户所指定的任务开始时间
Dependency	该作业所依赖的作业尚未完成

原因代码	详细说明
InvalidAccount	用户的 SLURM 账号无效
InvalidQoS	用户指定的 QoS 无效
PartitionTimeLimit	用户申请的时间超过该分区时间上限
QOSMaxCpuPerUserLimit	超过当前 QoS 用户最大 CPU 限制
QOSMaxGRESPerUser	超过当前 QoS 用户最大 GRES(GPU) 限制
Priority	存在一个或多个更高优先级的任务，该任务需要等待
ReqNodeNotAvail	所申请的部分节点不可用
Resources	暂无闲置资源，该任务需等待其他任务完成

Q36: 修改已经提交作业的 cpu 节点个数应该使用什么命令

A: 可以使用如下命令：

```
scontrol update jobid=** NumCPUs=*** NumNodes=***
```

其他 scontrol 相关的命令参数，可以使用 man scontrol 查看。

Q37: 如何一次提交多个小作业？

A: 可能有的用户在考虑如何一次提交多个小作业，又不会被系统 3 个同时运行作业的规定限制。可以尝试下面的脚本，注意最后那个 wait 不能丢。

```
#!/bin/bash
#SBATCH -J test # 作业名是 test
#SBATCH -p normal # 提交到 normal 分区
#SBATCH -n 4 # 提交 4 个 task
#SBATCH --cpus-per-task=1 # 每个 task 占用一个 cpu 核心
#SBATCH -t 5:00 # 任务最大运行时间是 5 分钟
#SBATCH -o test.out # 将屏幕的输出结果保存到当前文件夹的 test.out
srun -n 1 /usr/bin/sleep 100 &
srun -n 1 /usr/bin/sleep 100 &
srun -n 1 /usr/bin/sleep 100 &
srun -n 1 /usr/bin/sleep 100 &
wait
```

Q38: 修改已经提交作业的 cpu 节点个数应该使用什么命令？

A: 可以使用如下命令:

```
scontrol update jobid=** NumCPUs=*** NumNodes=***
```

其他 scontrol 相关的命令参数, 可以使用 `man scontrol` 查看。

四、应用计算相关

Q39: 使用 ansys 提交作业后, 报错: Licensed number of users already reached.

```
ANSYS LICENSE MANAGER ERROR:Failover feature 'ANSYS Academic Research CFD' specified i
n license preferences is not available.
Insufficient ANSYS Academic Research CFD licenses available.
Licensed number of users already reached.
Feature:      aa_r_cfd
License path: 1055@100.10.1.199:
FlexNet Licensing error:-4,132
For further information, refer to the FlexNet Licensing documentation,
available at "www.flexerasoftware.com".
```

A: 此报错为用户使用达到许可数量限制, 需等待空闲许可。

Q40: 提交 ansys 作业报错: Not enough Fluent

```
Unable to spawn node: license not available.
ANSYS LICENSE MANAGER ERROR:Not enough Fluent - HPC licenses 68/32.
```

A: 此报错为提交的作业并行核数超过了目前可用并行数。

Q41: 使用 pycharm 计算任务, 无法显示图形化报表。

A: 在程序中加入以下代码:

```
import matplotlib
matplotlib.use('Agg')
```

上述代码一定要添加在 `import matplotlib.pyplot` 之前, 否则无效。

Q42: lammps 编译的时候, 出现 `make[1]: mpicxx: Command not found`

A: mpicc 没有环境变量，需要使用命令 `module load intel` 加载 intel 模块。

Q43: 使用 pytorch 时，程序挂载在 gpu 节点上，但是没有用到显卡，并报错：`cuda_is_available false`

A: 检查脚本，是否有添加 `--gres` 参数，脚本没有此参数，则默认不分配 GPU 卡，建议添加显卡申请参数，`--gres`。

Q44: 平台 MATLAB 软件有哪些已经安装的工具箱？

A: 平台 MATLAB 已经安装工具箱如图

MATLAB	Version 9.7	(R2019b)
Simulink	Version 10.0	(R2019b)
5G Toolbox	Version 1.2	(R2019b)
Aerospace Blockset	Version 4.2	(R2019b)
Aerospace Toolbox	Version 3.2	(R2019b)
Communications Toolbox	Version 7.2	(R2019b)
Control System Toolbox	Version 10.7	(R2019b)
Curve Fitting Toolbox	Version 3.5.10	(R2019b)
DSP System Toolbox	Version 9.9	(R2019b)
MATLAB Coder	Version 4.3	(R2019b)
Optimization Toolbox	Version 8.4	(R2019b)
Signal Processing Toolbox	Version 8.3	(R2019b)
Simscape	Version 4.7	(R2019b)
Simscape Electrical	Version 7.2	(R2019b)
Simscape Multibody	Version 7.0	(R2019b)
Simulink Coder	Version 9.2	(R2019b)
Stateflow	Version 10.1	(R2019b)
Statistics and Machine Learning Toolbox	Version 11.6	(R2019b)

Q45: 计算报 `segmentation fault` 错误。

A: `segmentation fault` 是由于程序代码访问内存异常造成，常见于 C/C++ 程序代码中使用指针或数组访存操作。有些是访问了已经释放的内存，有些是访问了未经申请的内存，有些是数组越界访问，有些是使用的库版本不兼容，等等，原因各不相同。

一般的解决方法是，如果有源代码，则使用调试工具进行单步跟踪调试，逐步定位访存异常的代码，并逐步回溯，直到找到出现错误

的原因。如果没有源代码，则需要使用内核转储的分析工具对转储文件进行分析，分析工具的具体使用方法可到网上查找。

Q46: 如何在系统中使用 Infiniband 网络

A: 本系统配置了 Infiniband 网络，在使用 IntelMPI 和 OpenMPI 时，可以通过指定参数使用 Infiniband 网络。

使用 IntelMPI 时，请在运行 mpirun 时增加如下选项：

```
--genv I_MPI_FABRICS shm:dapl
```

使用 OpenMPI 时，请在运行 mpirun 时增加如下选项：

```
--mca btl self,sm,openib
```