WARNING!!

Before server usage description, after you access server,

you cannot execute any command (ex: ls, cd, etc.)

except permitted commands.

All commands entered to server are monitored in real time.

If forbidden commands are uncovered, that account will be erased immediately, and grade of that student will be influenced.
Index

• Copy files
• test_multicore_project.sh
• Help
• Show
• Compile
• Clean
• Submit
• Status
• Cancel
• Rank
• Error
Copy files

- Copy your files to server
- `scp [files] [StudentID@115.145.208.246:/.]`
- Example
Access Server

- Access Server
- ssh [StudentID@115.145.208.246]
- Example

```
jaeh@csl:~$ ssh 2012345678@SPBS00
2012345678@spbs00's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-72-generic x86_64)
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

패키지 93개를 업데이트할 수 있습니다.
0 업데이트는 보안 업데이트입니다.

*** 시스템을 다시 시작해야 합니다 ***
Last login: Mon May 1 05:31:13 2017 from 115.145.211.192
2012345678@SPBS00:~$```
test_multicore_project.sh

• Basic command. All instructions are started by using this command.

• If commands which are not started by using this command are found, that account will be erased.

• Example (Auto complete when press keyboard tab button after type “test_”)

```
2012345678@SPBS00:~$ test_multicore_project.sh --submit

=====================================================================
== JobID: 473, USERNAME: 2012345678, Running life ==
=====================================================================
2012345678@SPBS00:~$
```
Help

- Print possible command option list

- Enter command option which is not posted help list is prohibited.

- Example

```
$ test_multicore_project.sh --help

Usage: test_multicore_project.sh --submit
Description: Put your binary 'life' in run queue

Usage: test_multicore_project.sh --status [JOB ID]
Description: Print status of your binary 'life'

Usage: test_multicore_project.sh --cancel [JOB ID]
Description: cancel your binary 'life' at run queue

Usage: test_multicore_project.sh --rank
Description: Display total rank

Usage: test_multicore_project.sh --error
Description: Display error list

Usage: test_multicore_project.sh --compile
Description: Execute 'make'. Compile your files with skeleton files

Usage: test_multicore_project.sh --clean
Description: Execute 'make clean'

Usage: test_multicore_project.sh --show
Description: Execute 'ls'
```
Show

- File which is being in your directory check

- life.c, list.*, lcgrand.*, setup.h, etc. are in skeleton directory ("cd skeleton" is ban)

- Example

```
2012345678@SPBS00:~$ test_multicore_project.sh --show
==================================== Your File List ===============================
Makefile  game.c  game.h  skeleton
====================================
2012345678@SPBS00:~$
```
Compile

- Same execution about command execution “make”

- All logs are registered in real time. If commands which need not compile are found, that account is erased immediately (will compare Makefile submitted at Sequential project to this)

- Example

```
2012345678@SPBS00:--$ test_multicore_project.sh --compile
================================================================== Compile Start =================================================================================
gcc -c lcgrand.c lcgrand.h
gcc -c list.c list.h
gcc -c game.c game.h
gcc -o life life.c lcgrand.o list.o game.o
================================================================== Compile End =================================================================================
2012345678@SPBS00:--$
```
Clean

• Same execution about command execution “make clean”

• All logs are registered like Compile option

• Example

```
2012345678@SPBS00:~$ test_multicore_project.sh --clean
====================================== Make Clean Start ======================================
rm *.o
rm *.gch
====================================== Make Clean End ======================================
2012345678@SPBS00:~$
```
Submit

• Push “life” execution file to server queue *(If name is not “life”, this is not pushed to queue)*

• Wait in queue until your turn (PD/pending state)

• Execution state can be checked by using printed JobID (check next page)

• Example

```
2012345678@SPBS00:~$ test_multicore_project.sh --submit

== JobID: 473, USERNAME: 2012345678, Running life ==

2012345678@SPBS00:~$
```
Status

- Check current status of “life” execution file

- ST: Current state
  (CD: completed, R: running, CA: canceled TO: Timeout PD: pending, F: Fail)

- Time: Toal execution time of your “life” execution time

- Example

```
2012345678@SPBS01:--$ test_multicore_project.sh --status 473

.IOBSID PARTITION NAME USER ST TIME NODES NODELIST(REASON)
 473   R930  life  20123456  CD  0:14   1 SPBS01

ID    : 473
Task   : life
User   : 2012345678
Status : COMPLETED
Enqueued: (Day) 2017-05-01 / (Time) 04:28:59
Executed: (Day) 2017-05-01 / (Time) 04:28:59
Finished: (Day) 2017-05-01 / (Time) 04:29:13
ASGNODE : SPBS01
NODENUM  : 1

2012345678@SPBS01:--$  
```
Cancel

- Erase “life” execution file at server queue or Cancel execution of your file
- If JobID is not yours, cannot delete
- Example

```
2012345678@SPBS00:--$ test_multicore_project.sh --cancel 474
================================
== Your job 474 is canceled ==
================================
2012345678@SPBS00:--$
```
• Print all students execution time

• Example

```
2012345678@SPBS00:~$ test_multicore_project.sh --rank
======================================
[Rank] [ID]       [Time]
 1   2012345678   12.484291
======================================
2012345678@SPBS00:~$
```
Error

- Output file Incorrect / Timeout / Cancelled cases are printed

- Example

```bash
2012345678@SPBS00:~$ test_multicore_project.sh --error
======================================================================
[ID]   [Invalid Output files / Error state]
2012345678 Timeout
======================================================================
2012345678@SPBS00:~$
```