

Project 1: NPShell



NP TA 孟宇

10/20 18:20

Project 1 Deadline

Demo: 10/22 Thu.

General Info

- We will announce our [Bitbucket](#) organization.
- We will announce the [nplinux](#) account.
 - NP projects should run on NP servers.
 - Any abuse of NP server will be recorded.
 - Don't leave any zombie processes in the system.

Project 1: Info

- You are **HIGHLY** encouraged to publish your questions on Project 1 討論區
- You can contact TAs by email: np@cgilab.nctu.edu.tw (Mails sent to other addresses will NOT be replied)
- TA hours (Thursday: 15:00 ~ 17:00) on **10/8**, **10/15** will be held at **EC511**
- TAs will **NOT** debug for you.

Project 1: Submission

- Create a directory named as your student ID, put all files into the directory.
- You **MUST** use **GNU Make** to build your project and compile your source code into one executable named **npshell**. The executable and Makefile should be placed at the top layer of the directory. We will use this executable for demo.
- You are **NOT** allowed to demo if we are unable to compile your project with a single make command.
- Upload only your code and Makefile. Do NOT upload anything else (e.g. noop, removetag, test.html, **.git**, **__MACOSX**)
- zip the directory and upload the .zip file to the E3 platform.

ATTENTION! We only accept .zip format

4. **zip** the directory and upload the .zip file to the E3 platform

ATTENTION! We only accept .zip format

e.g.

Create a directory 0856053, the directory structure may be:

0856053

```
|—— Makefile  
|—— shell.cpp  
|—— shell.h
```

zip the folder 0856053 into 0856053.zip and upload 0856053.zip onto E3

G. We take plagiarism seriously.

All projects will be checked by a cutting-edge plagiarism detector.

You will get zero points on this project for plagiarism.

Please don't copy-paste any code from the internet, this may be considered plagiarism as well.

Protect your code from being stolen.

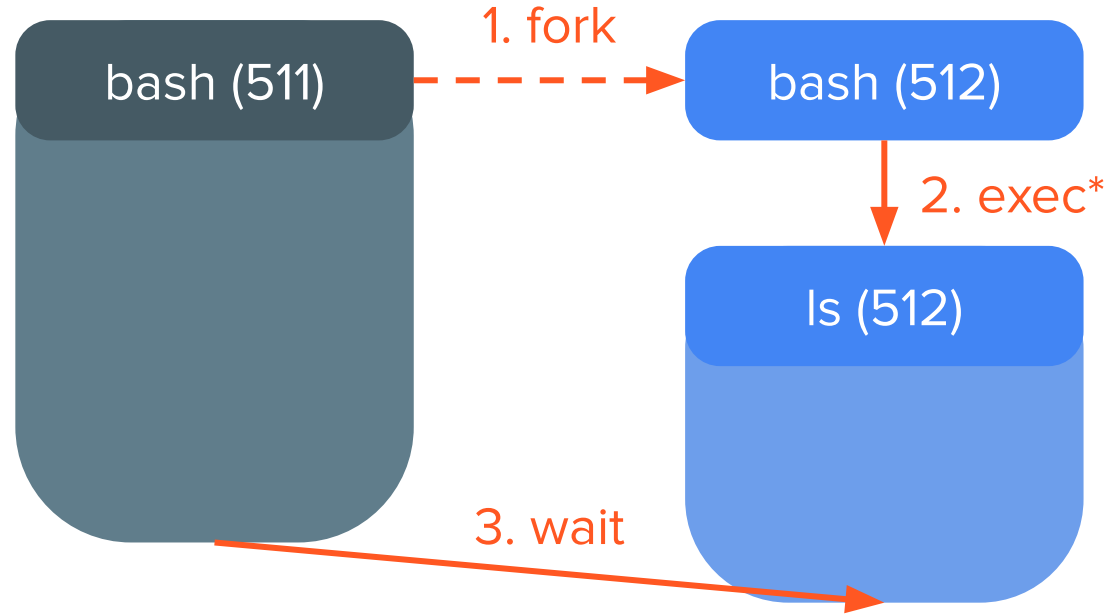
Project 1: Demo

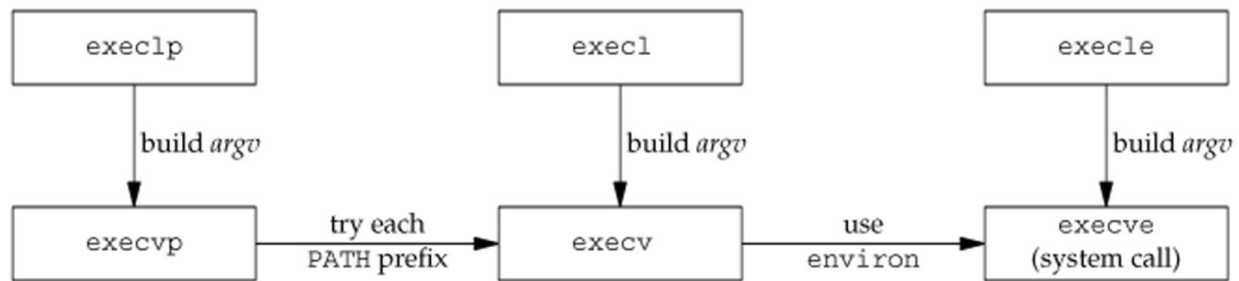
- 10/22 Thu. 13:10 ~ 21:30
- We will announce demo slots 2~3 days before.
- Tasks:
 - (correct format and compile)
 - QA?
 - Pass np-basic test cases
 - Pass np-hard test cases
 - Implement 1 or 2 extra functions with limited time

Implementation

Process Lifecycle: fork-exec-wait

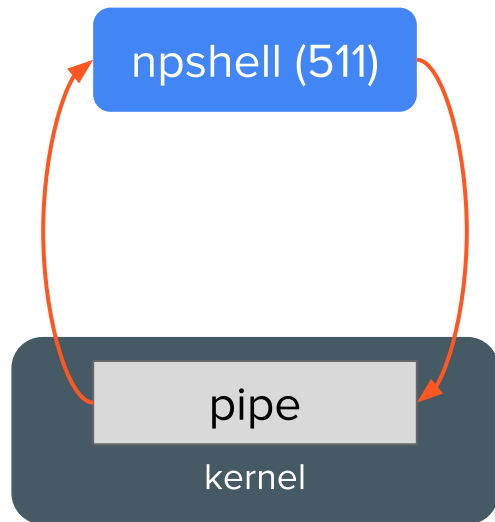
1. Creation: fork
2. Execution exec*
3. Termination: waitpid



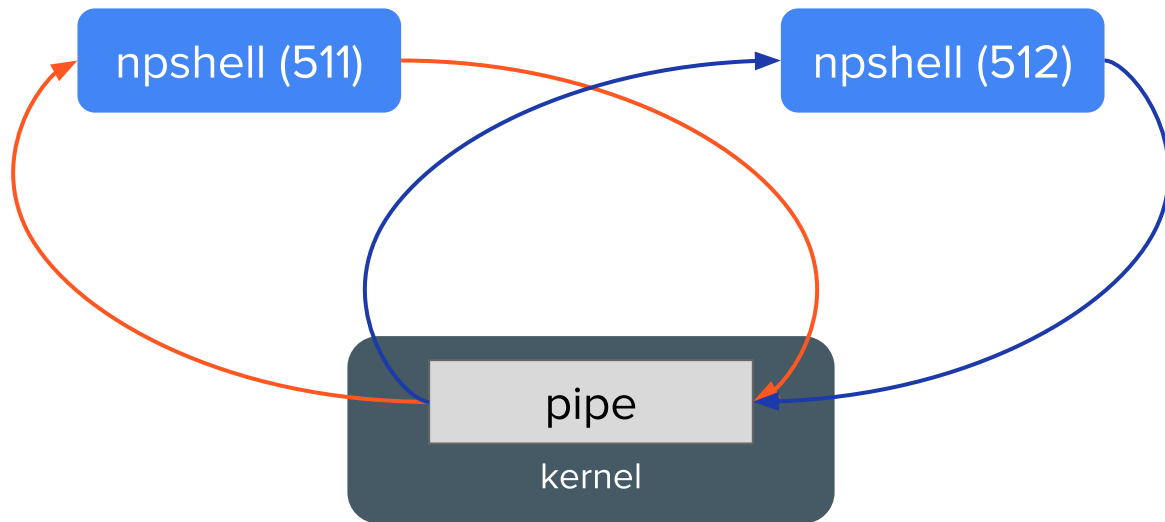


Pipe

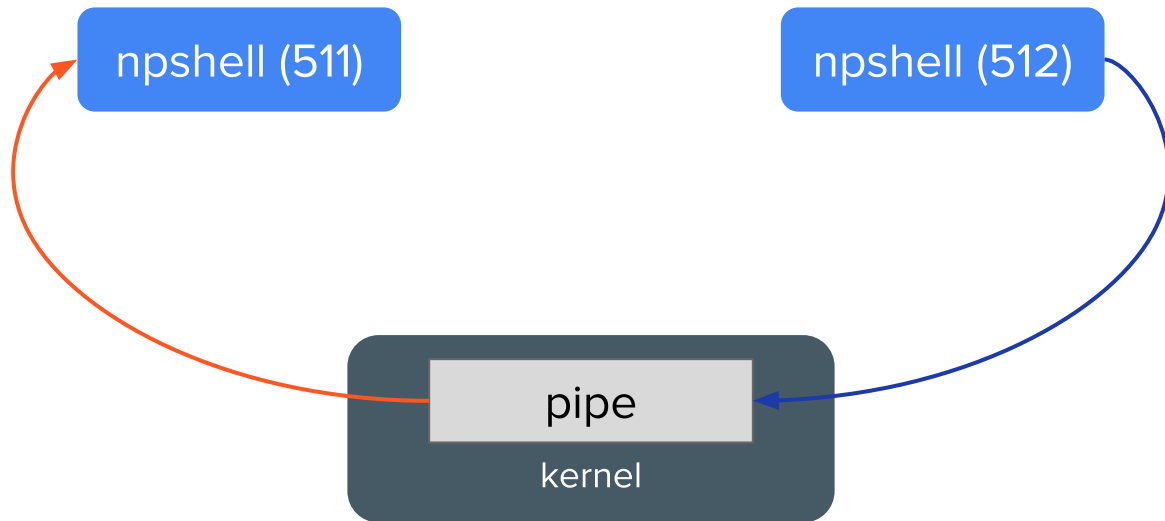
0. pipe



1. fork

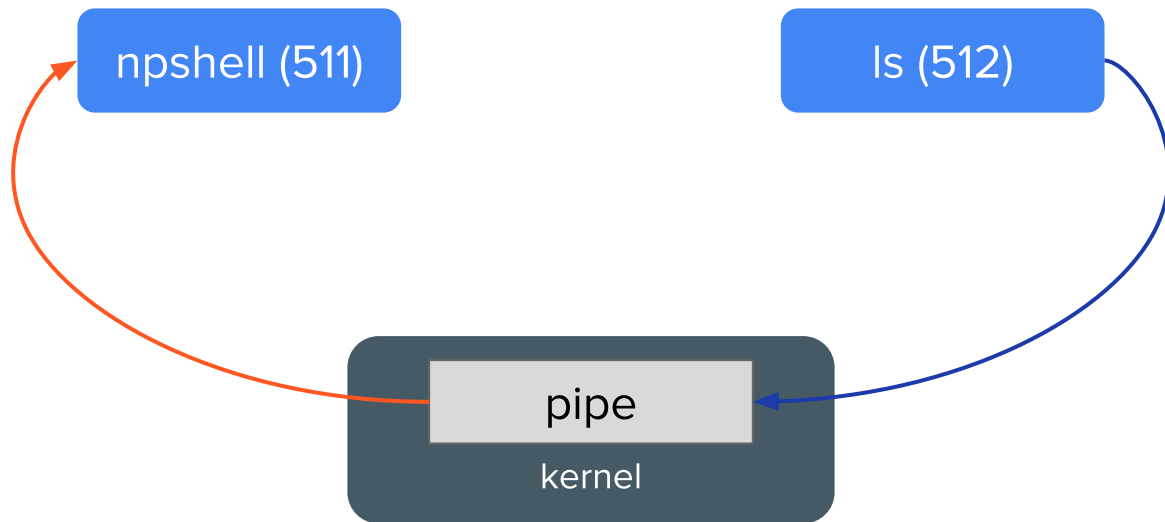


2. close



(because pipe is half-duplex)

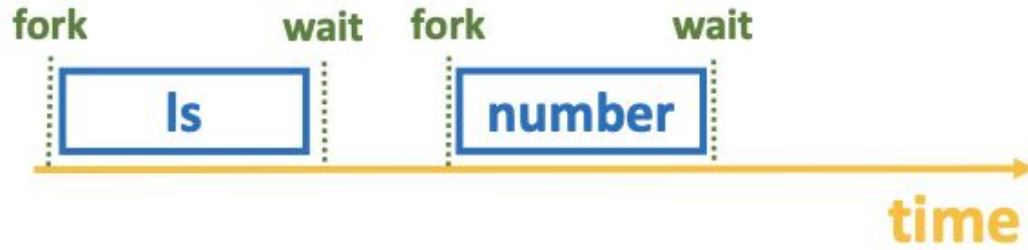
3. exec



Issues

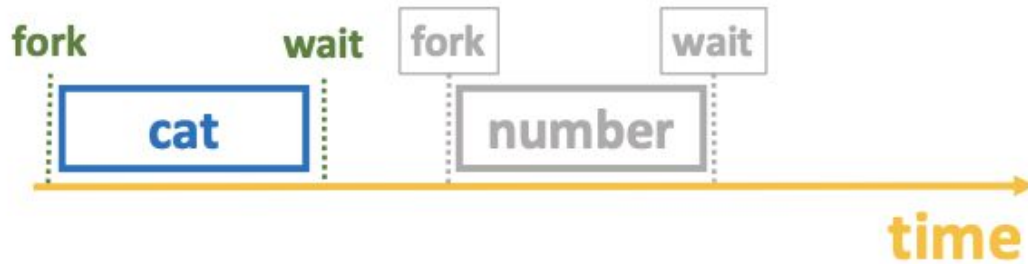
Impl 1 : Wait for each child

% ls | number



Problem 1 : Unable to process large data

```
% cat largeFile.txt | number % cat largeFile.txt | 1  
% number
```



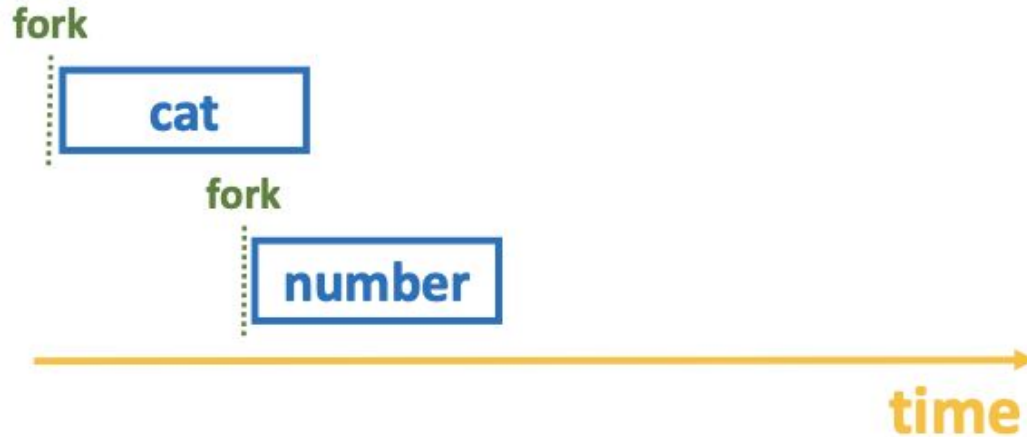
The process will hang forever !

Notes

- Please don't store data into temporary files.

Impl 2 : Don't wait for processes

```
% cat largeFile.txt | number
```



Problem 2 : % ordering

% ls
% bin test.html

% fork



%



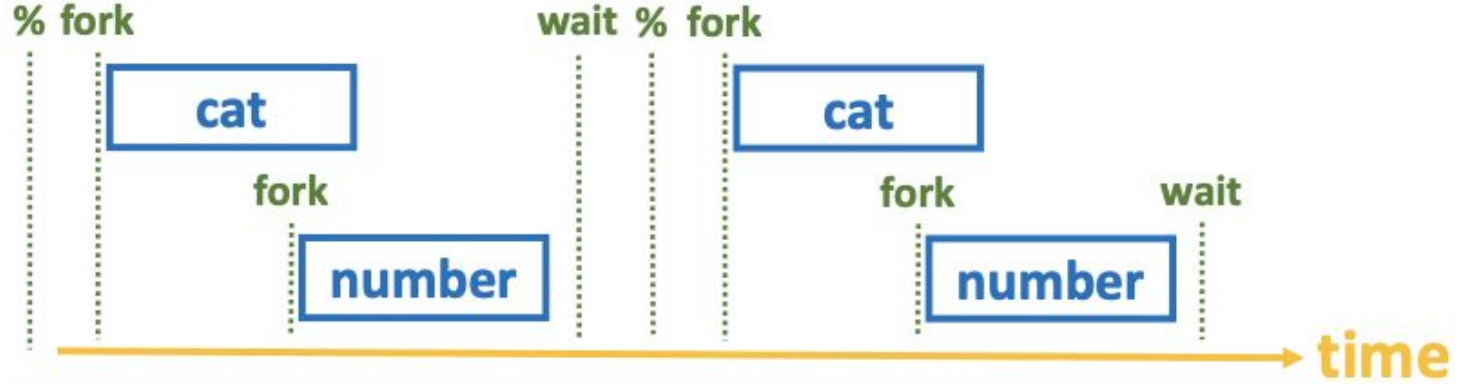
time

Correct:

% ls
bin test.html
%

Impl 3 : Wait for a line if not pipeN

```
% cat largeFile.txt | number  
% cat largeFile.txt | 1  
% number
```

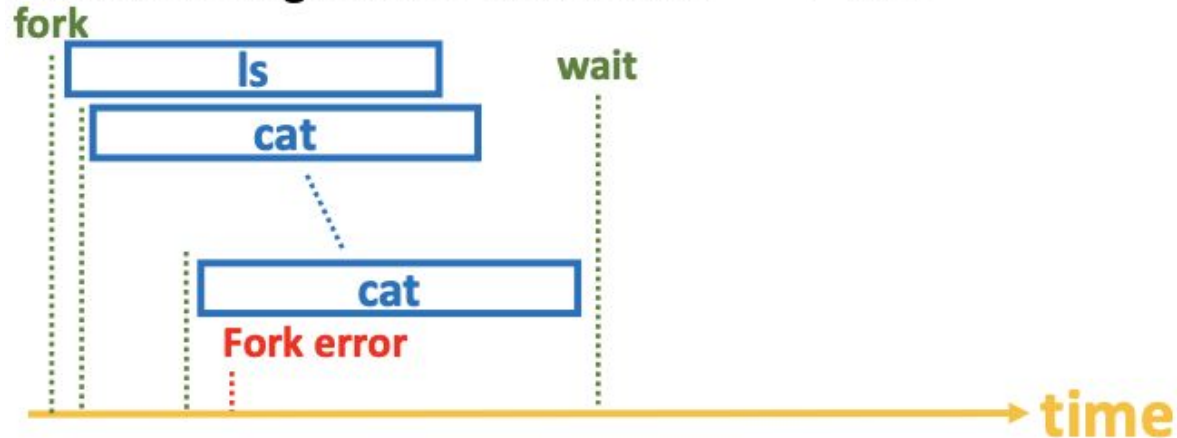


Problem 4 : Process limitation

ls | cat | cat | cat | cat cat | cat

Process limit is **512** on NP server.

Fork error might occur around the 510th cat.



More Problems...

不在此爆雷

Hint

- Functions you may use
 - fork
 - pipe
 - dup, dup2
 - exec*
 - wait, waitpid
- Handle failure
 - See man page for more information
- Debug
 - gdb
 - lsof

Enjoy the project!