Process Assignment – 2

Team Emertxe

WAP to avoid a child become zombie with out blocking the parent.

Assignment - 2

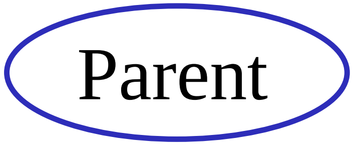
Requirements

- ✓ Create a child process avoid it become a zombie.

Assignment - 2

Requirements

- ✓ Create a child process avoid it become a zombie.

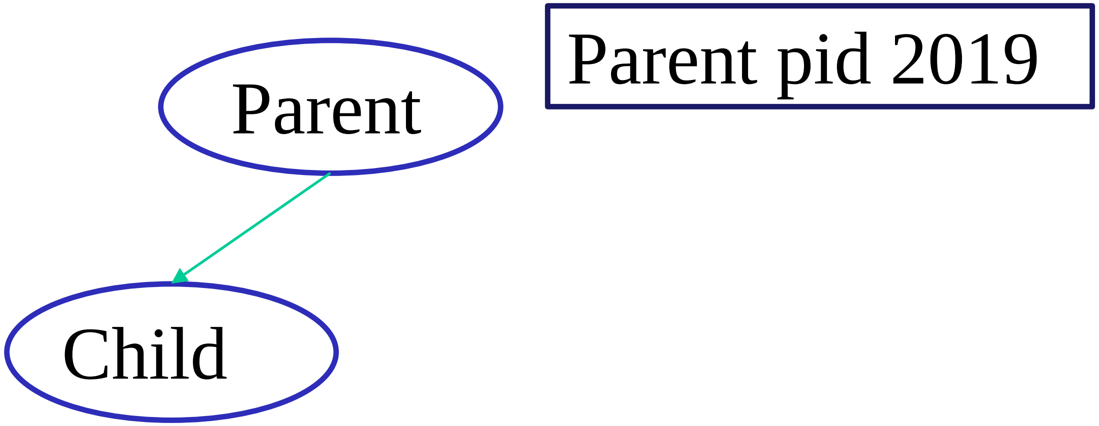


Parent

Assignment - 2

Requirements

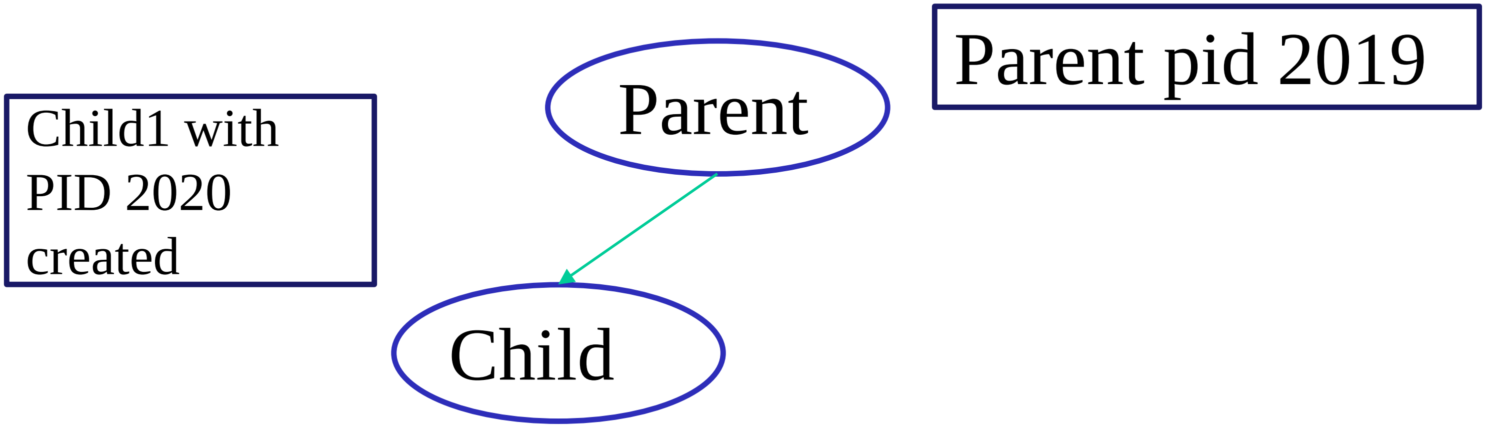
✓ Create a child process avoid it become a zombie.


Parent pid 2019

Assignment - 2

Requirements

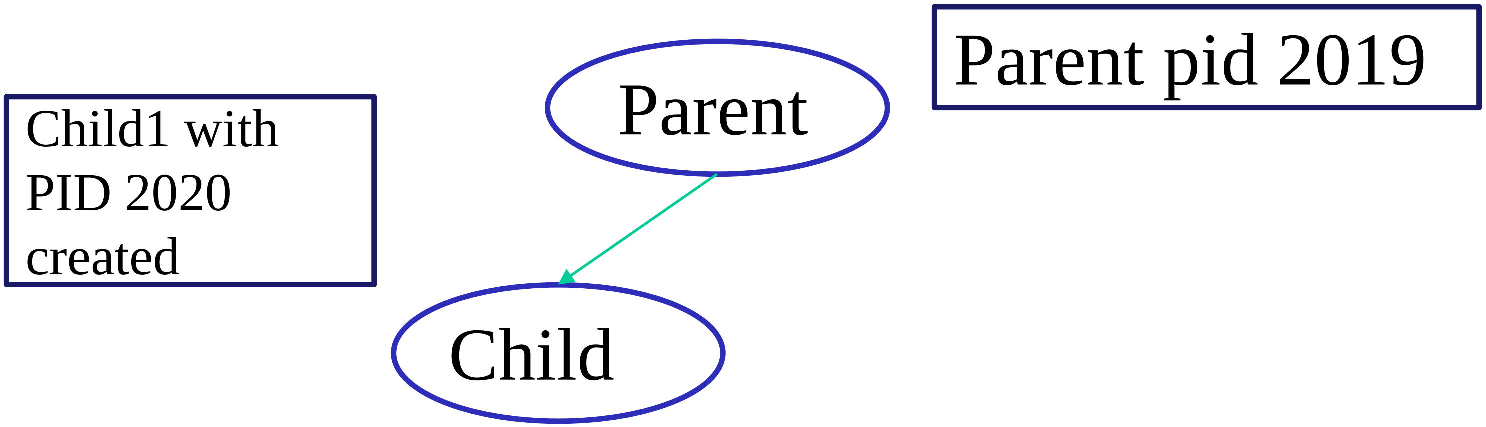
- ✓ Create a child process avoid it become a zombie.



Assignment - 2

Requirements

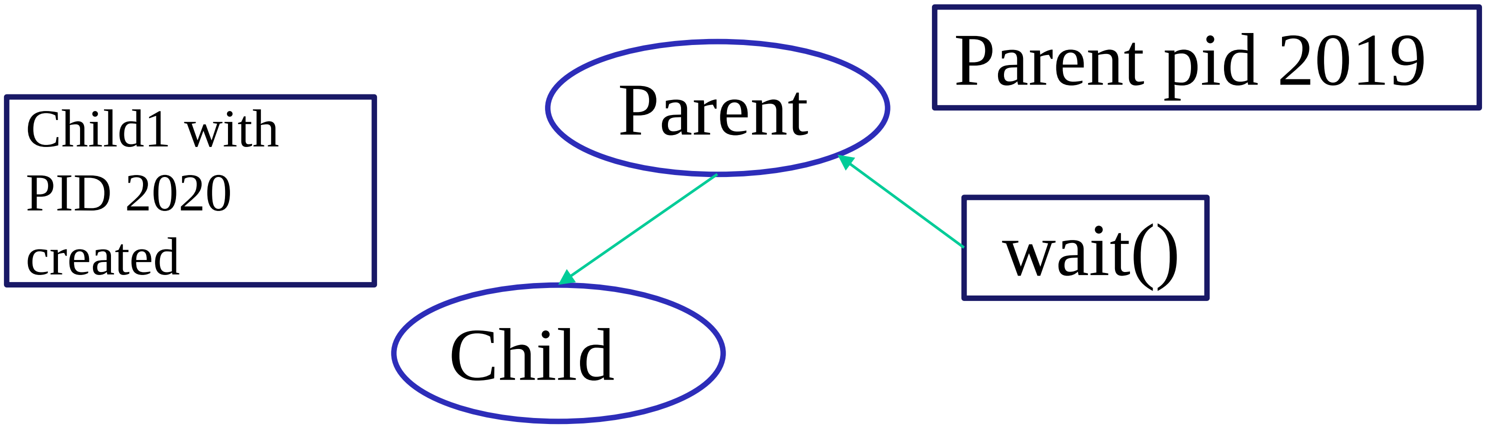
- ✓ To avoid zombie we need to call wait(),



Assignment - 2

Requirements

- ✓ To avoid zombie we need to call wait(), but this block parent until child terminates.



Parent pid 2019

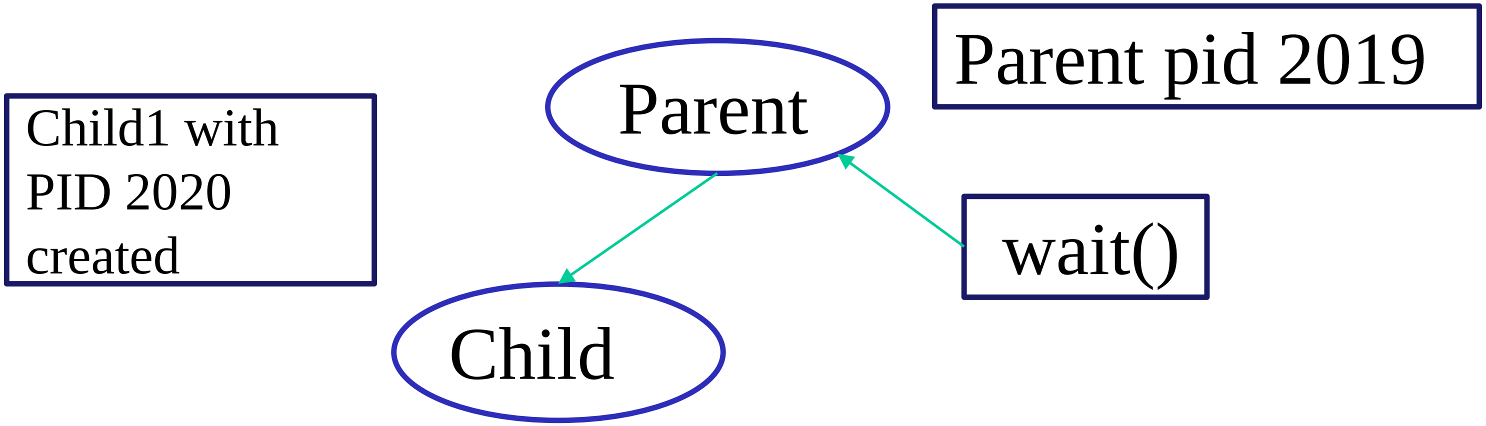
Child1 with PID 2020 created

wait()

Assignment - 2

Requirements

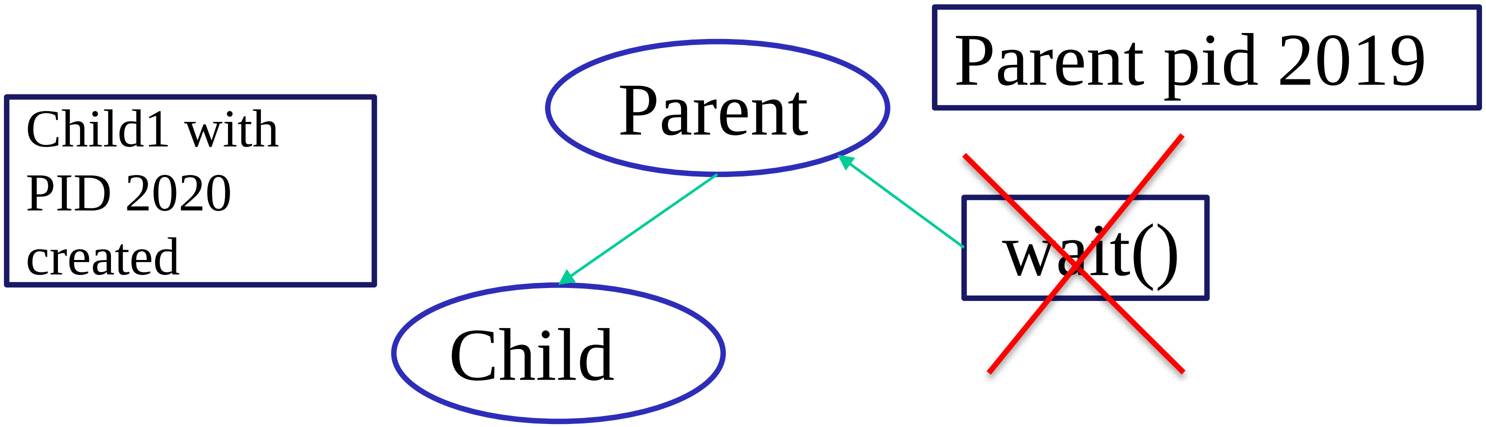
- ✓ But this block parent until child terminates.



Assignment - 2

Requirements

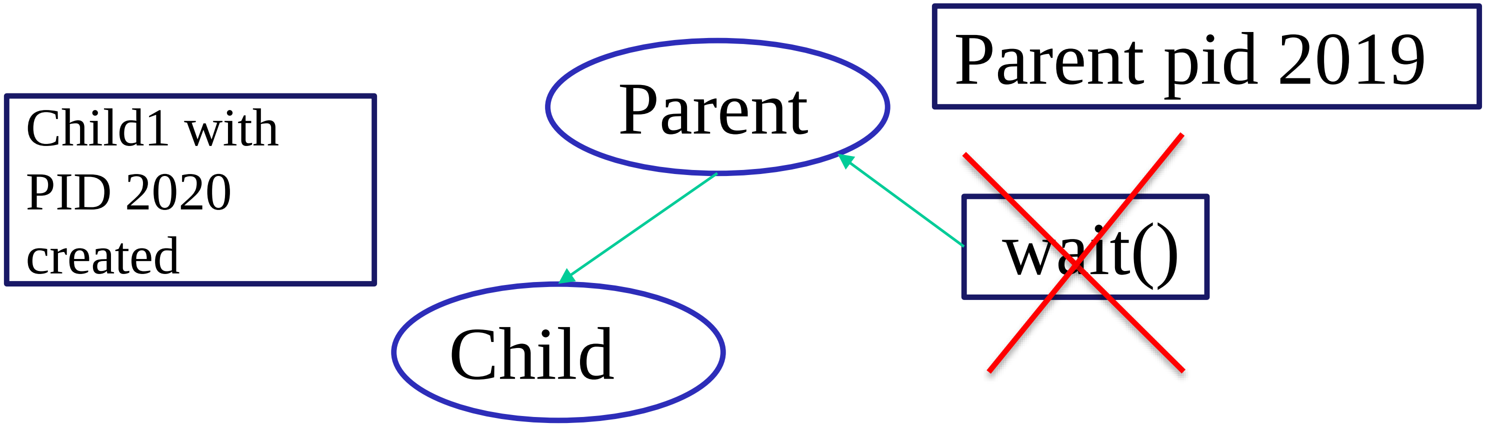
- ✓ But this block parent until child terminates.



Assignment - 2

Requirements

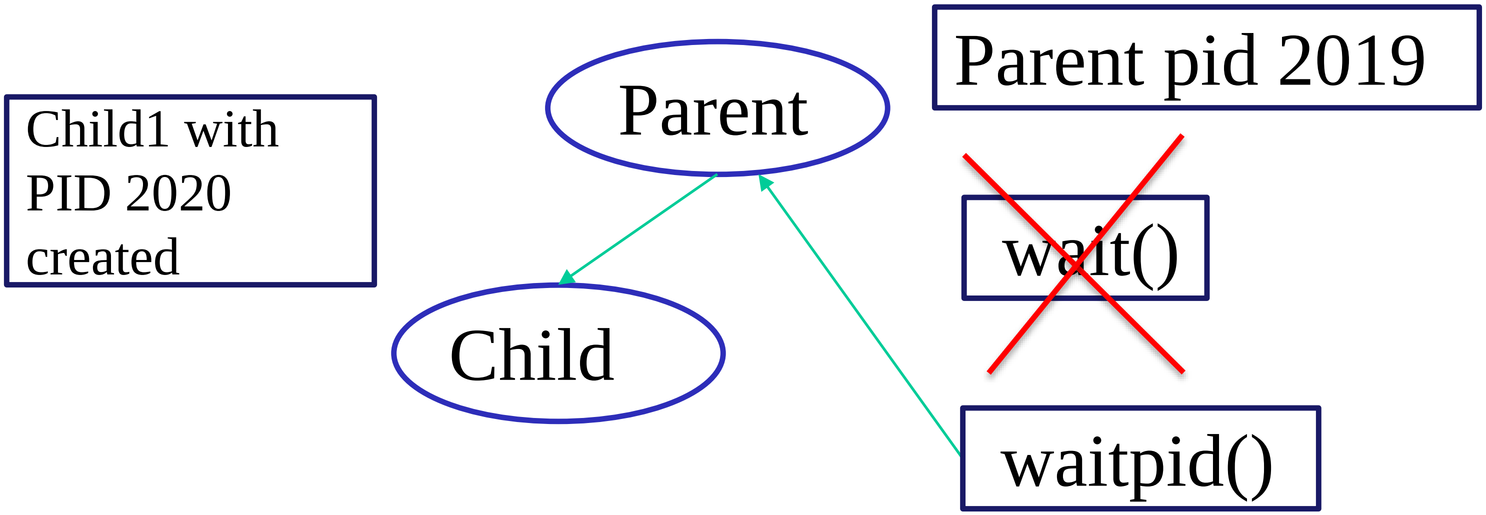
- ✓ So we need to use waitpid() with proper arguments (Read man page).



Assignment - 2

Requirements

- ✓ So we need to use waitpid() with proper arguments (Read man page).



Assignment - 2

Requirements

- ✓ When child is working parent has to continuously print some message.
- ✓ When ever child terminates parent has to print child terminated and print exit status of child process.

Assignment - 2

Sample execution

1. ./nonblock_wait

   ```
   A child created with pid 2020
   ```

Assignment - 2

Sample execution

1. ./nonblock_wait

   ```
   A child created with pid 2020
   parent is running
   ```

Assignment - 2

Sample execution

1. ./nonblock_wait

```
A child created with pid 2020
parent is running
parent is running
```

Assignment - 2

Sample execution

1. ./nonblock_wait

```
A child created with pid 2020
parent is running
parent is running
parent is running
```

Assignment - 2

Sample execution

1. ./nonblock_wait

```
A child created with pid 2020
parent is running
parent is running
parent is running
Child 2020 terminated normally with exit status 0
```

Assignment - 2

Sample execution

1. ./nonblock_wait

```
A child created with pid 2020
parent is running
parent is running
parent is running
Child 2020 terminated normally with exit status 0
parent terminating
```

Assignment - 2

Pre-requisites

✓ Knowledge about system calls, How to read and understand ‘man pages’.

Assignment - 2

Pre-requisites

- ✓ Knowledge about system calls, How to read and understand ‘man pages’.
- ✓ Good knowledge about processes & zombie process.

Assignment - 2

Pre-requisites

- ✓ Knowledge about system calls, How to read and understand ‘man pages’.
- ✓ Good knowledge about processes & zombie process.
- ✓ Working of fork & wait system call.

Assignment - 2

Pre-requisites

- ✓ Knowledge about system calls, How to read and understand ‘man pages’.
- ✓ Good knowledge about processes & zombie process.
- ✓ Working of fork & wait system call.

Objective

- ✓ To understand different states of a process.