



Process Assignment – 1

Team Emertxe





Create a scenario to make zombie process become orphan, print status of each state.

Assignment - 1

Requirements

- ✓ Create a child process and print the status of that process in Parent process.

Assignment - 1

Requirements

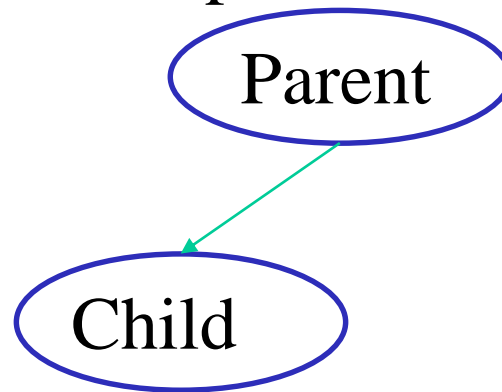
- ✓ Create a child process and print the status of that process in Parent process.

Parent

Assignment - 1

Requirements

- ✓ Create a child process and print the status of that process in Parent process.

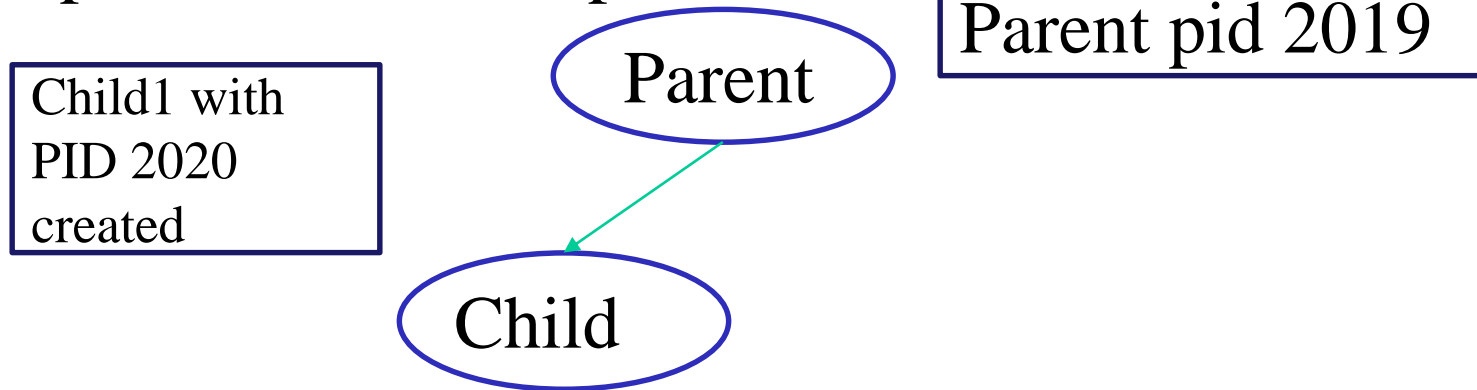


Parent pid 2019

Assignment - 1

Requirements

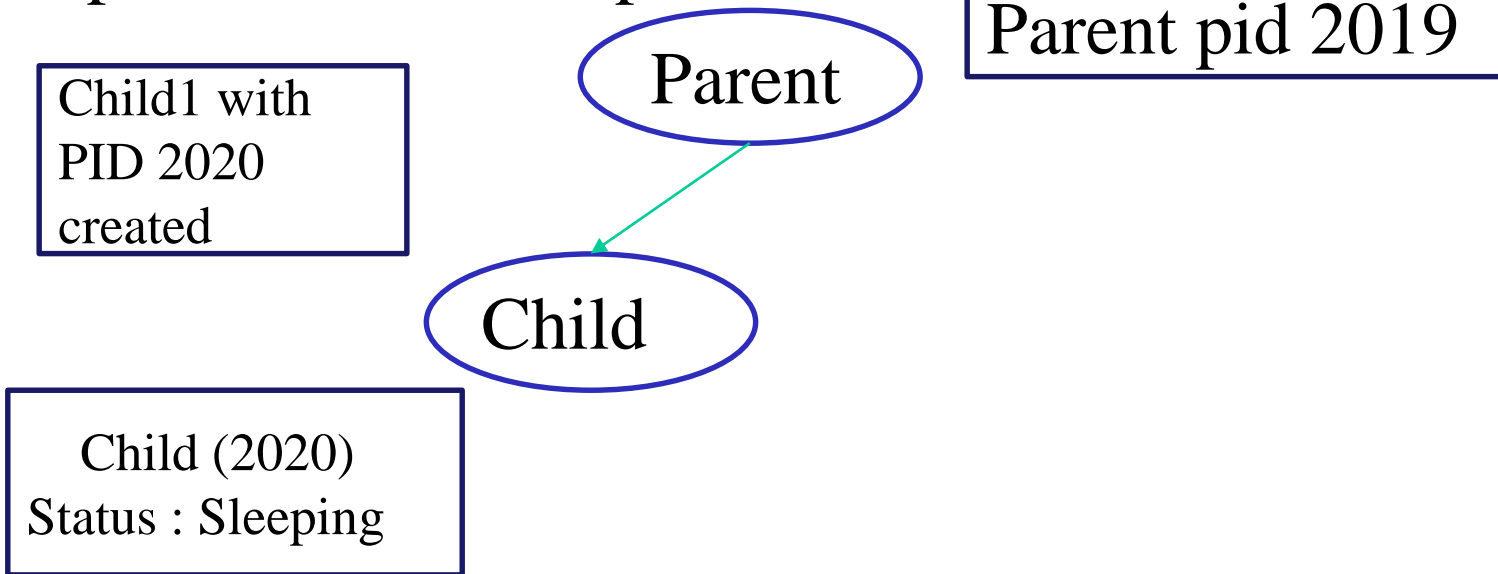
- ✓ Create a child process and print the status of that process in Parent process.



Assignment - 1

Requirements

- ✓ Create a child process and print the status of that process in Parent process.



Assignment - 1

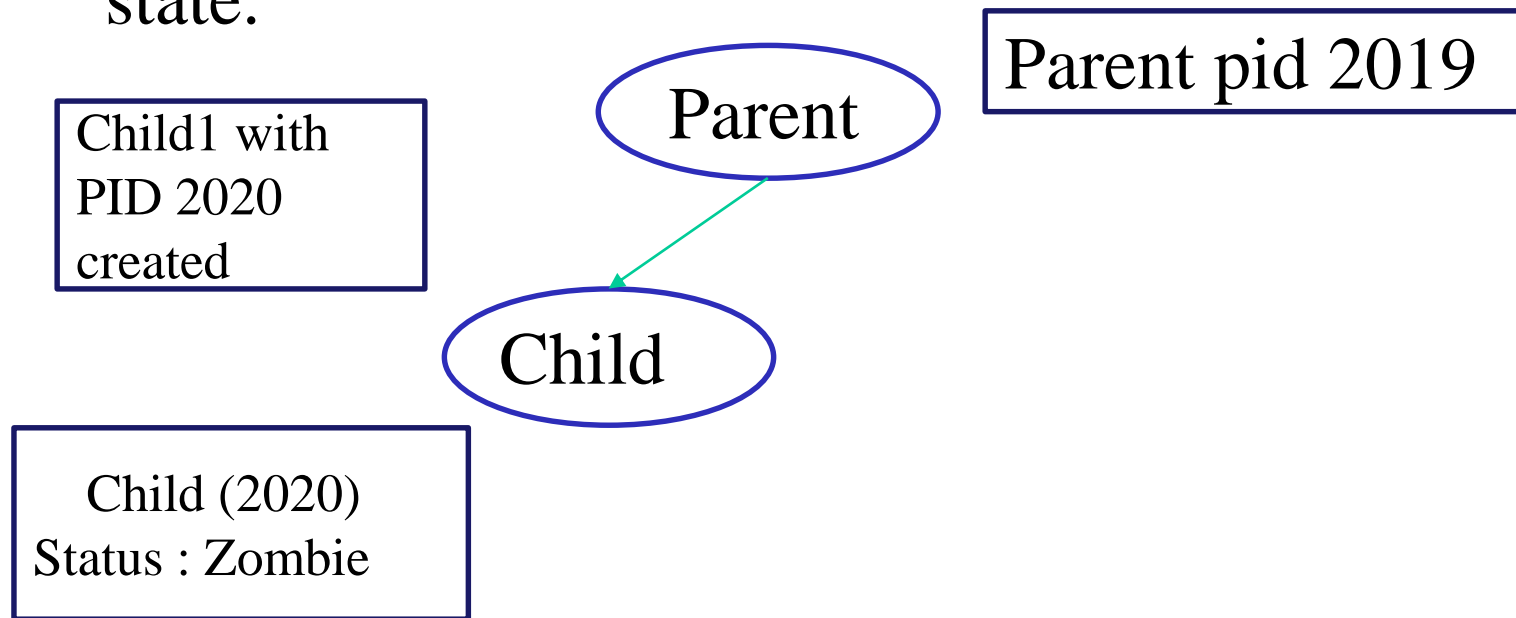
Requirements

- ✓ After some time print status that process is zombie state.

Assignment - 1

Requirements

- ✓ After some time print status that process is zombie state.



Assignment - 1

Requirements

- ✓ After some time (After parent's execution completed) print zombie process cleared by init.

Assignment - 1

Requirements

- ✓ After some time (After parent's execution completed) print zombie process cleared by init.
- ✓ Now the question arrives is in which process this message should be printed? Both process got terminated.

Assignment - 1

Requirements

- ✓ After some time (After parent's execution completed) print zombie process cleared by init.
- ✓ Now the question arrives is in which process this message should be printed? Both process got terminated.
- ✓ So one more child process should be created before terminating the parent process.

Assignment - 1

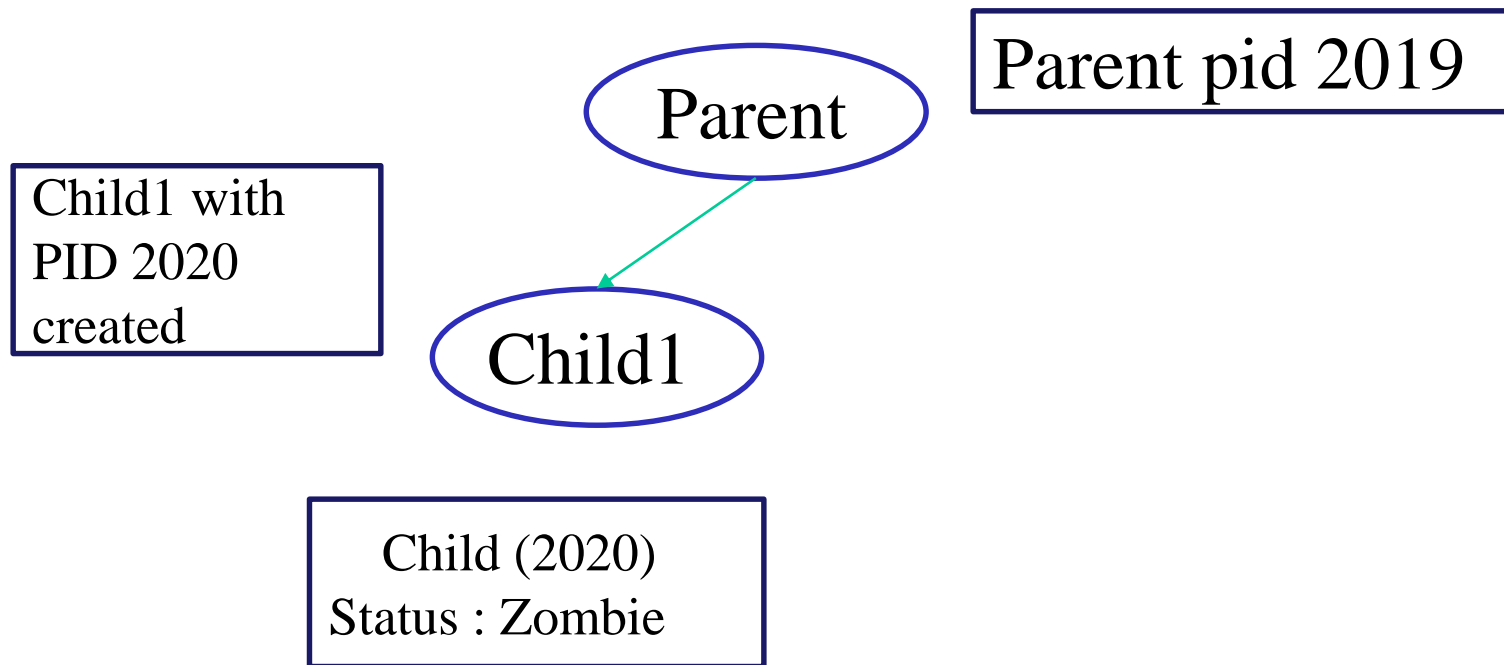
Requirements

- ✓ After some time (After parent's execution completed) print zombie process cleared by init.
- ✓ Now the question arrives is in which process this message should be printed? Both process got terminated.
- ✓ So one more child process should be created before terminating the parent process.
- ✓ After parent got terminated both child and child2 becomes orphan.

Assignment - 1

Requirements

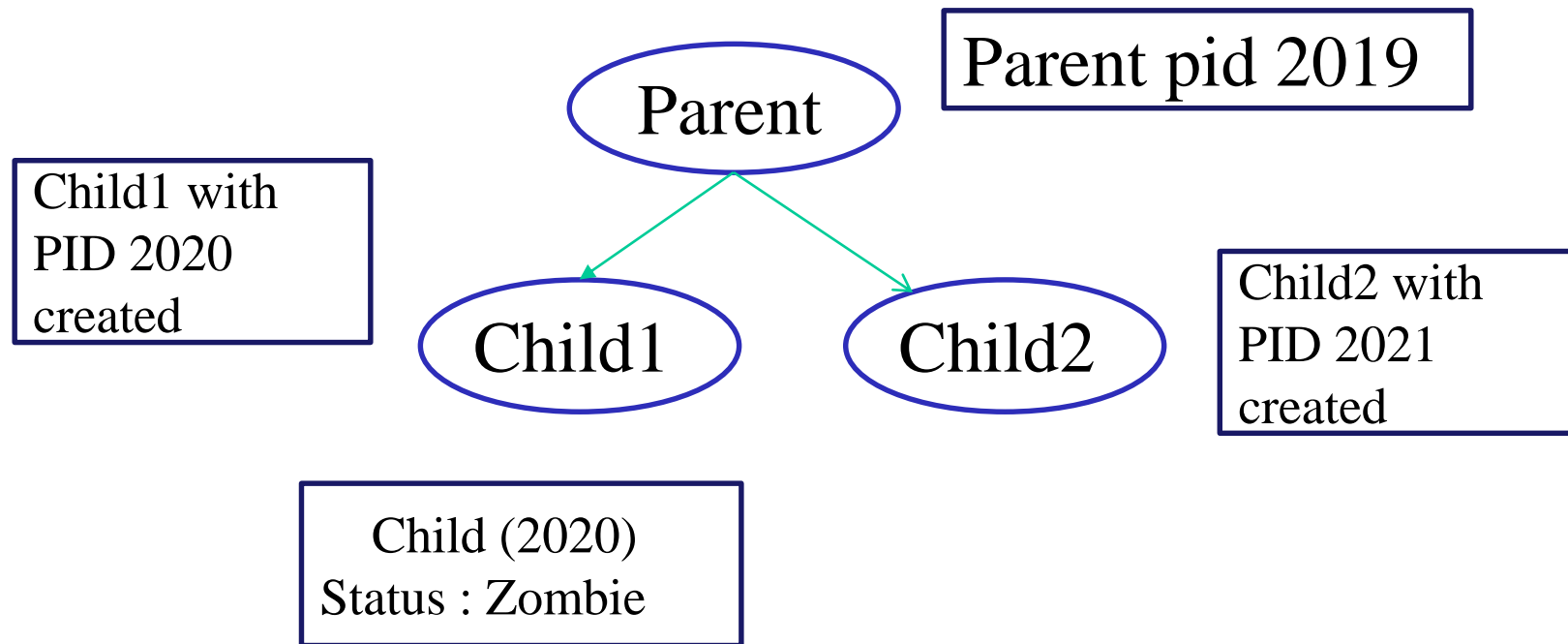
- ✓ Creating child2.



Assignment - 1

Requirements

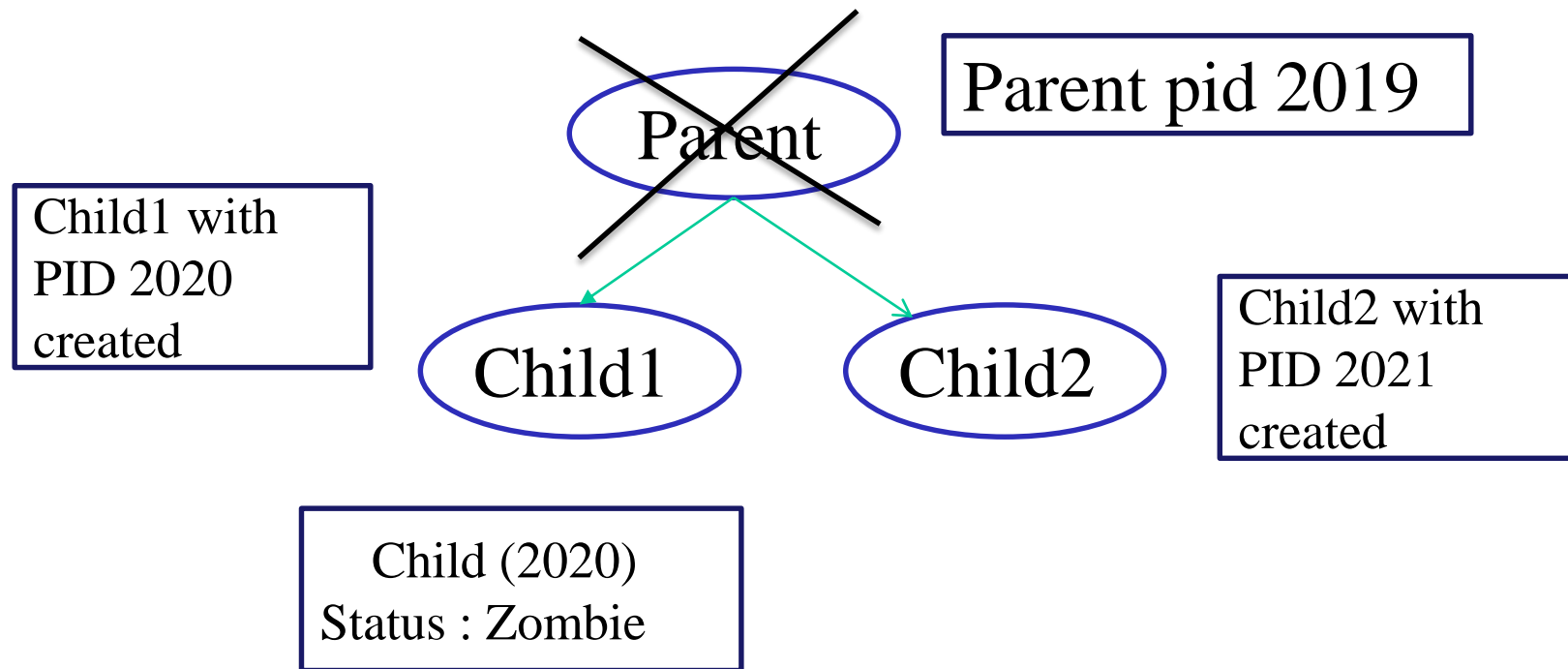
- ✓ After creating child2.



Assignment - 1

Requirements

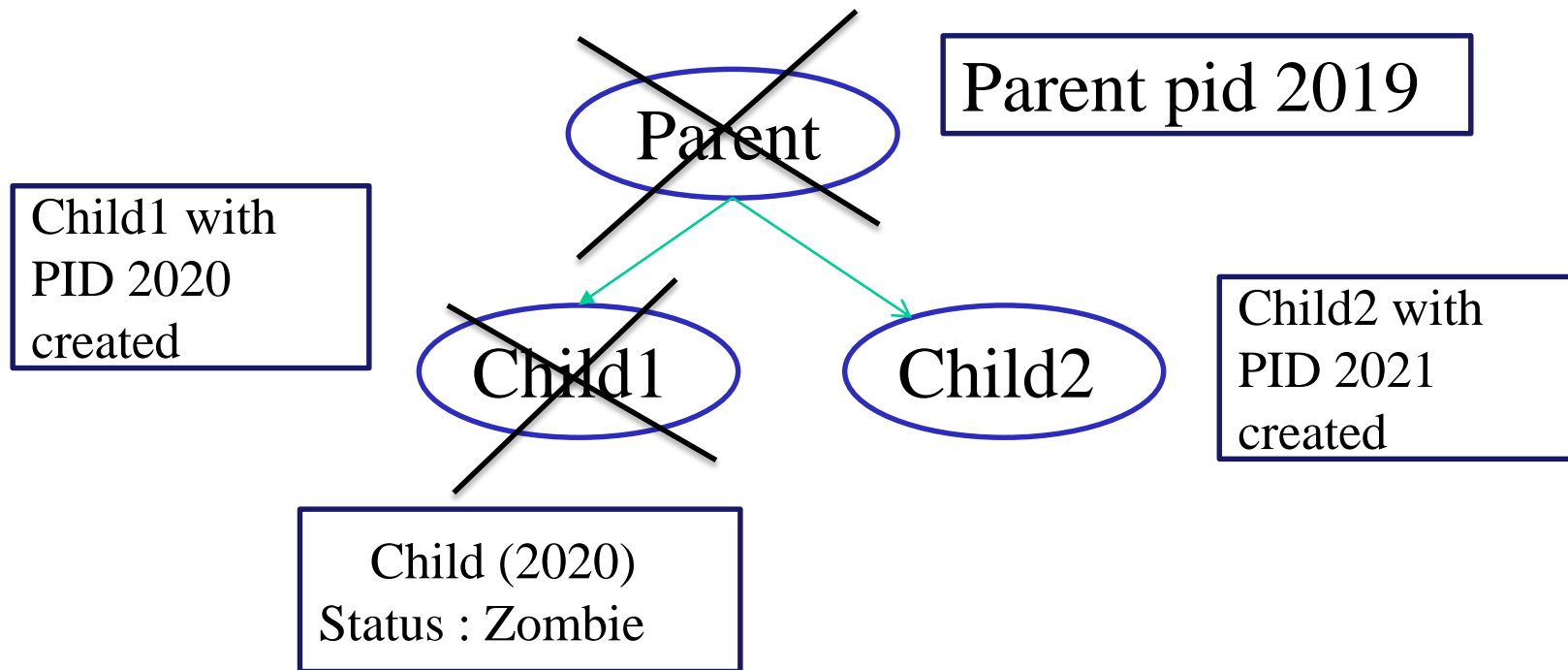
- ✓ After Parent terminated.



Assignment - 1

Requirements

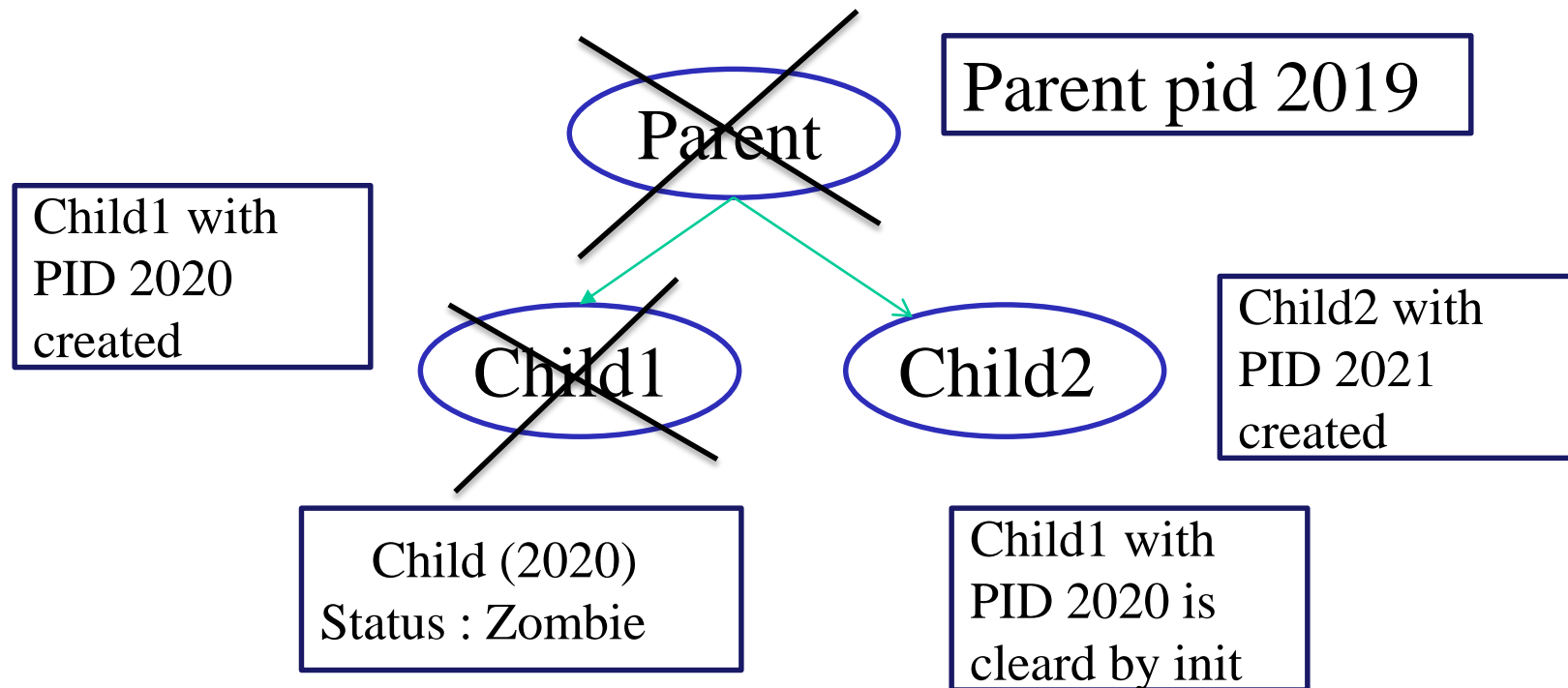
✓ After Parent terminated.



Assignment - 1

Requirements

- ✓ After Parent terminated.



Assignment - 1

Requirements

- ✓ How to get the current status of any process? And how to check whether the process is alive or its terminated?

Assignment - 1

Requirements

- ✓ How to get the current status of any process? And how to check whether the process is alive or its terminated?
- ✓ To print status of any process use help of /proc/<pid>/status file. For eg: if child pid is 2020, open file proc/2020/status and print first 3 lines.

Assignment - 1

Requirements

- ✓ How to get the current status of any process? And how to check whether the process is alive or its terminated?
- ✓ To print status of any process use help of /proc/<pid>/status file. For eg: if child pid is 2020, open file proc/2020/status and print first 3 lines.
- ✓ If that file is not available means that process is cleared.

Assignment - 1

Sample execution

1. ./zomb_orph

Assignment - 1

Sample execution

1. ./zomb_orph

A child created with pid 2020

Name: ./zomb_orph

State: S (sleeping)

Assignment - 1



Sample execution

1. ./zomb_orph

A child created with pid 2020

Name: ./zomb_orph

State: S (sleeping)

Name: /zomb_orph (After some time)

State: Z (zombie)

Assignment - 1



Sample execution

1. ./zomb_orph

A child created with pid 2020

Name: ./zomb_orph

State: S (sleeping)

Name: /zomb_orph (After some time)

State: Z (zombie)

Process 1234 cleared by init

Assignment - 1

Pre-requisites

- ✓ Knowledge about system calls, How to read and understand 'man pages'.

Assignment - 1

Pre-requisites

- ✓ Knowledge about system calls, How to read and understand 'man pages'.
- ✓ Good knowledge about processes, zombie and orphan.

Assignment - 1

Pre-requisites

- ✓ Knowledge about system calls, How to read and understand 'man pages'.
- ✓ Good knowledge about processes, zombie and orphan.
- ✓ Working of fork system call.

Assignment - 1

Pre-requisites

- ✓ Knowledge about system calls, How to read and understand 'man pages'.
- ✓ Good knowledge about processes, zombie and orphan.
- ✓ Working of fork system call

Objective

- ✓ To understand different states of a process.