Assignment 2 - Part 1: A Star Wars Process

Fall 2024

Due: Thursday, Sep 26, 11:59 pm

Coding Task 1: Complete the program A2_p.c

- Complete the program to spawn 4 child processes each representing different Star Wars character:
    - Luke – adjusts shield power by 25%
    - Han – adjusts shield power by 20%
    - Chewbacca – adjusts shield power by 30%
    - Leia – adjusts shield power by 15%.
- Each child process should:
    - Print “<character name>: Adjusting shield power”
    - Increment the shield power by the specified percentage.
    - Print new shield level: “<character name>: Shield power level now at <x>%”.
- The parent process must:
    - Check if process creation fails, and handle errors appropriately.
    - Wait for all child processes to finish their task before it exits.

Your output should look like this.

```
Millennium Falcon: Initial shield power level: 50%

Luke: Adjusting shields...
Luke: Shield power level now at 75%
Han: Adjusting shields...
Han: Shield power level now at 70%
Chewbacca: Adjusting shield...
Chewbacca: Shield power level now at 80%
Leia: Adjusting shield...
Leia: Shield power level now at 65%

Millennium Falcon: Final shield power level: 50%

May the forks be with you!
```

Task 2: Explain the following about the program A2_p.c:

1) Why does the initial and final value of the shield power remain unchanged?
2) Why does the order of output change every time you run the program?

Provide a clear, concise, and detailed explanation.

Task 3: Answer the following:

Imagine you're running a popular website like an online store during a big sale. Hundreds of users are placing orders at the same time. How would using separate processes help ensure that every customer can successfully place their order?

Deliverables:

- A zipped folder named A2_part1 submitted to D2L dropbox that contains:
    - Your source code file A2_p.c.
    - A pdf file for Task 2, Task 3, and a screenshot showing your code A2_p.c works as expected.
    - A README.txt file
- Available on Github inside the folder where you added me as a collaborator.

Total Points (100)

- Code runs and works as expected – Task 1: 60 points
- Clear and detailed explanation – Task 2: 20 points
- Clear and detailed explanation – Task 2: 15 points
- Available on GitHub: 5 points