

L2: Klaster obliczeniowy

Wymagane flagi kompilacji: `-Wall -fsanitize=address,undefined`

Napisz program symulujący klaster obliczeniowy.

Proces-rodzic tworzy N dzieci, gdzie N jest jedynym argumentem przyjmowanym przez program.

Rodzic zleca pierwszemu dziecku pracę, wysyłając sygnał `SIGUSR1`. Dziecko po otrzymaniu sygnału rozpoczyna pracę w pętli, śpi losową ilość czasu z przedziału 100-200ms, po czym dodaje jeden do lokalnego licznika i wypisuje `{PID}: {licznik}\n` na `stdout`.

Jeśli rodzic otrzyma sygnał `SIGUSR1` to wysyła sygnał `SIGUSR2` do obecnie pracującego dziecka i `SIGUSR1` do następnego dziecka (na zasadzie round robin). Dziecko, które otrzyma sygnał `SIGUSR2` zatrzymuje swoją pętlę i oczekuje na sygnał `SIGUSR1` aby wznowić działanie.

Po otrzymaniu sygnału `SIGINT` rodzic przekazuje ten sam sygnał wszystkim dzieciom. Po otrzymaniu sygnału `SIGINT` dzieci zapisują (wykorzystując wywołanie systemowe) swój licznik do pliku `PID.txt` i kończą działanie. Rodzic czeka na zakończenie działania wszystkich dzieci i kończy pracę.

Etapy:

- 4 p. Proces-rodzic tworzy dzieci. Proces-dziecko wypisuje swój PID i kończy działanie. Rodzic czeka na zakończenie działania wszystkich dzieci i kończy pracę.
- 5 p. Rodzic zleca pierwszemu dziecku pracę, wysyłając sygnał `SIGUSR1`. Dziecko po otrzymaniu sygnału wykonuje pracę w pętli.
- 4 p. Obsługa sygnału `SIGUSR1` przez rodzica, obsługa sygnału `SIGUSR2` przez dzieci.
- 3 p. Obsługa zakończenia działania programu za pomocą sygnału `SIGINT`. Zapis liczników do pliku przez dzieci.

Etap	1	2	3	4	Suma
Punkty	4	5	4	3	16
Wynik					