CN Lab 12-1-2022: Assignments-1 : Signals

1. Circular Signalling – Two way

Processes P1 , P2 , P3 , P4 will form a circle by knowing their pids of previous and next process of it, using a Message Q as explained in below steps. (steps must be followed as specified only.)

- P2 gets executed first, and it sends a message containing its pid as : p2pid with type value 2.
- P1 gets executed , it receives message and finds p2pid from message and sends SIGUSR1 to P2. P1 notes that its next (right side) process is P2 and its pid is p2pid.
- P2 catches the signal and **it finds out, who has sent that signal to it.** (that is P2 should find the pid of P1 after receiving signal from P1).(Hint : **sigaction()** .. can be used for this). Now P2 also notes that its previous(left) process is P1(i.e. p1pid).
- P3 gets executed, and it sends a message containing its pid as : p3pid with type value 3.
- P2 receives the message from msq and it notes that its next (right side) process is P3 and its pid is p3pid.
- **P2 sends a message containing P1's pid as : <u>type = 1</u>, content = p1pid.**
- P2 sends a signal SIGUSR1 to P3.
- P3 catches the signal and **it finds out, who has sent that signal to it.** (that is P3 should find the pid of P2 after receiving signal from P2).(Hint : **sigaction()** .. can be used for this)). Now P3 also notes that its previous(left) process is P2 (i.e. p2pid).
- P4 gets executed, and it sends a message containing its pid as : p4pid with type value 4.
  - (Note that in the message queue msq, there are two messages now, one is type=1, p1pid, another is type=4, p4pid)
- P3 receives the message from msq of type=4 , and it notes that its next (right side) process is P4 and its pid is p4pid.
- P3 sends a signal SIGUSR1 to P4.
- P4 catches the signal and **it finds out, who has sent that signal to it.** (that is P4 should find the pid of P3 after receiving signal from P3).(Hint : **sigaction()** .. can be used for this)). Now P4 also notes that its previous(left) process is P3 (i.e. p3pid).
- P4 receives a message from msq (there is one message left in msq) . P4 notes out from that message that its next (right side) process is P1, and sends a SIGUSR1 to P1 (ie. p1pid).
- P1 catches the signal and **it finds out, who has sent that signal to it.** (that is P1 should find the pid of P4 after receiving signal from P4).(Hint : **sigaction()** .. can be used for this)). Now P1 also notes that its previous(left) process is P4 (i.e. p4pid).
- Now the circular signalling should as : P1 signals to P2, after receiving that signal, P2 signals to P3, then P3 to P4, and P4 to P1.
- This circular signalling of P1 → P2 → P3 → P4 → P1 should happen **3 times**.
- Now Reverse circular signalling should be as : P1 signals P4, P4 to P3, P3 to P2, P2 to P1.
- This reverse circular signalling of P1 → P4 → P3 → P2 → P1 should happen **3 times**.
- All the processes are to be stopped.
  (You may use SIGUSR1 for circular signalling and SIGUSR2 for reverse circular signalling)
  <u>Submission link will be sent later</u>.