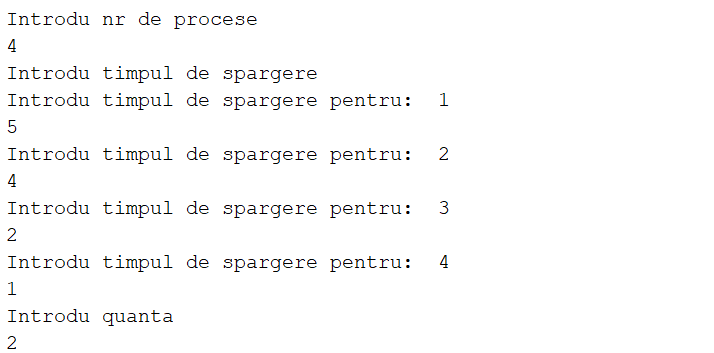
**PROIECT**

**Round Robin** este un algoritm foarte popular de programare CPU.Acest algoritm programează procesele pe baza timpului lor de sosire, a timpului de spargere și a cuantumului timpului CPU.

**De ce avem nevoie de un astfel de algoritm ?**

Realizam mai multe taskuri pe calculatorul nostrum simultan, descarcăm un film de pe internet, ascultam muzică și programam, toate în același timp. Dar calculatorul nostru nu are suficiente nuclee, pentru a realiza mai multe cereri. CPU-ul nostru folosește schimbarea cererii pentru a depăși problema lipsei nucleelor. Noi consideram că toate procesele se execută simultan, dar ele nu sunt realizate in paralel. Algoritmul de programare a procesorului rulează un proces pentru un timp, îl oprește și rulează un alt proces .Această schimbare de cereri este atât de rapidă încât se pare că toate procesele rulează simultan.

Consideram procesul:



Acum avem ceva numit "Coada de așteptare" și "Coada de rulare" în care se adaugă procesele. Când timpul de sosire al procesorului se potrivește cu timpul procesorului, procesul este gata de procesare și, prin urmare, adăugat la coada de așteptare. De exemplu, procesul p1 are timpul de sosire 0, Deci, în prezent, coada noastră de gata va avea procesul p0.

**Coada de așteptare:P1**

**Coada de rulare:P1**

După 2 unități de timp, procesele p2 și p3 vor fi gata pentru procesare, astfel încât le putem adăuga la coada de asteptare. Deoarece p1 este procesat pentru quanta 2, timpul său de spargere va fi redus la 3.

Dupa cum putem observa p1 nu este încă complet, așa că trebuie să îl adăugăm din nou la coada de asteptare.

După încă două unități de timp, p2 se va opri din executare și p3 va intra în coada de rulare.

**Coada de așteptare:P2,P3,P1.**

**Coada de rulare:P1,P2.**

La rularea lui p3, deoarece timpul de spargere p3 este 2, acesta se va executa complet și nu va mai fi adăugat la coada de asteptare din nou.

**Coada de așteptare:P3,P1,P4,P2**

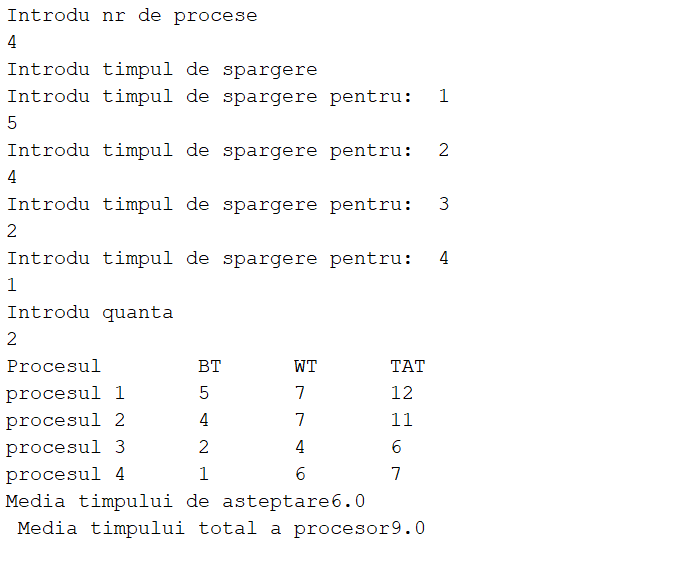
**Coada de rulare:P1,P2,P3.**

Există încă un punct de observat, procesul p4 a avut un timp de spargere de 1 unitate. Deci execuția se va întâmpla pentru 1 unitate, nu pentru 2 unități, ceea ce reprezintă timpul de spargere .

**Coada de așteptare:P1,P4,P2**

**Coada de rulare:P1,P2,P3.**

În acest fel, toate procesele se execută până când au timpul lor de spargere de 0 unități.



Calina Florin

-Grupa 222-

>Informatica<