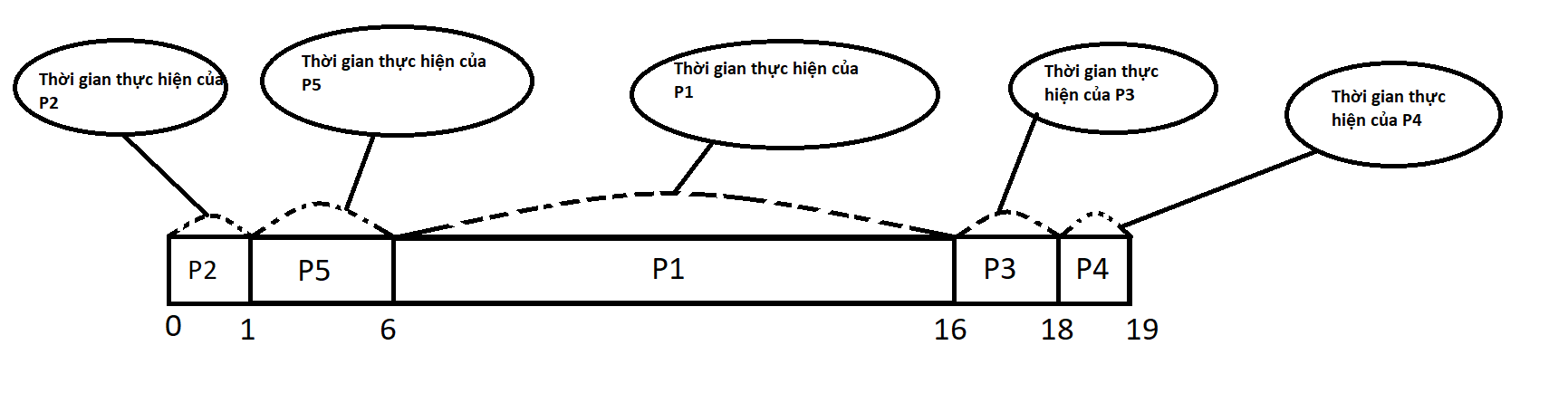
|  |  |  |
| --- | --- | --- |
| Tiến Trình (Process) | Thời gian thực hiện (Burst Time) | Mức Ưu Tiên (Priority) |
| P1 | 10 | 3 |
| P2 | 1 | 1 |
| P3 | 2 | 4 |
| P4 | 1 | 5 |
| P5 | 5 | 2 |

**Lý Thuyết**

Mỗi tiến trình gắn với 1 mức ưu tiên   
Tiến trình sẽ thực hiện từ mức ưu tiên nhỏ tới lớn nhất



Từ đây tính được thời gian chờ của mỗi tiến trình

P2 thực hiện đầu tiên nên thời gian chờ = 0 và thời gian thực hiện là 1  
P5 phải chờ P2 thực hiện xong mới được thực hiện nên thời gian chờ = thời gian thực hiện của P2 và bằng 1

Tương tự P1 phải đợi P5 P2 thực hiện xong mới có thể thực hiện nên thời gian chờ bằng tổng thời gian thực hiện của P5 và P2 = 1 + 5 = 6

Tương tự thời gian chờ P3 = Tổng thời gian thực hiện của(P5 P2 P1) = 1 + 5 + 10 = 16   
Tương tự thời gian chờ P4 = 1 + 5 + 10 +2 = 18

Thời gian chờ trung bình sẽ là trung bình cộng thời gian chờ của P1 P2 P3 P4 P5 = (0+1+6+16+18)/5=8.2

**Ý Tưởng Giải Thuật**

Dùng 1 biến số nguyên để lưu số tiến trình nhập vào Vd biến n

Dùng 3 mảng để lưu giá trị của **số tiến trình(process)**  (ví dụ p[]), **thời gian thực hiện(burst time)** (ví dụ bt[]) và **mức ưu tiên(priority)**  (ví dụ pr[])

Dùng vòng lặp for để nhập và in ra giá trị của **thời gian thực hiện(burst time)** và **mức ưu tiên(priority)**

**for(i=0;i<n;i++)**

**{**

**cout<<"\nP["<<i+1<<"]\n";**

**cout<<"Thoi Gian Thuc Hien: ";**

**cin>>bt[i];**

**cout<<"Muc Uu Tien: ";**

**cin>>pr[i];**

**p[i]=i+1; //Chua So Tien Trinh**

**}**

Dùng giải thuật sắp xếp chọn(selection sort) để sắp xếp thời gian thực hiện, mức ưu tiên, số tiến trình theo thứ tự tăng dần

**for(i=0;i<n;i++)**

**{**

**int min=i;**

**for(j=i+1;j<n;j++)**

**{**

**if(pr[j]<pr[min])**

**min=j;**

**}**

**//Đổi chỗ thời gian thực hiện,mức ưu tiên, tiến trình,**

**tg=pr[i];**

**pr[i]=pr[min];**

**pr[min]=tg;**

**tg=bt[i];**

**bt[i]=bt[min];**

**bt[min]=tg;**

**tg=p[i];**

**p[i]=p[min];**

**p[min]=tg;**

**}**

Tiếp theo là tính **thời gian chờ của từng tiến trình** và **tổng thời gian chờ**

Dùng 1 mảng để chứa giá trị **của thời gian chờ của từng tiến trình** (ví dụ wt[]) và 1 biến chứa **giá trị tổng thời gian chờ** (ví dụ )

Công thức tính **thời gian chờ của từng tiến trình** là

Thời gian chờ của tiến trình này bằng **tổng thời gian thực hiện của các tiến trình còn lại**

Vì tiến trình đầu tiên đi vào thực hiện luôn nên thời gian chờ sẽ = 0 nên t gán luôn thời gian chờ trung bình của tiến trình = 0

(vd wt[0] = 0)

Dùng vòng lặp for 2 nhau để tính **thời gian chờ trung bình** và **tổng thời gian chờ**

**for(i=1;i<n;i++) // bắt đầu từ 1 vì đã gán thời gian chờ của tiến trình thứ nhất wt[0] = 0**

**{**

**wt[i]=0;**

**for(j=0;j<i;j++) wt[i]+=bt[j]; // thời gian chờ của tiến trình thứ i = tổng thời gian thực hiện(burst time) của các tiến trình trước**

**tong+=wt[i]; //tính tổng thời gian chờ**

**}**

Tiếp theo là tính **thời gian chờ trung bình** của các tiến trình

Dùng tiếp 1 biến để chứa giá trị của **thời gian chờ trung bình** (vd wt\_tb)

**Thời gian chờ trung bình** = **tổng thời gian thực hiện** chia cho **số tiến trình**

wt\_tb = tong/n;

Cuối cùng là dùng vòng lặp for để đưa ra **thời gian chờ** và **thời gian chờ trung bình**

**for(i=0;i<n;i++)**

**{**

**cout<<"\nP["<<p[i]<<"] "<<" Thoi gian thuc hien: "bt[i]<<" Thoi gian cho: "<<wt[i];**

**}**

**cout<<"\n\nThoi Gian Cho Trung Binh: = "<<wt\_tb;**