

نحوه کار و پیاده سازی Multi-level queue scheduling algorithm

نحوه کار ای الگوریتم به آن صورت است که چندین Queue مختلف داریم برای انتقال پردازش ها به داخل آن ها. بین Queue ها اولویت داریم و همچنین داخل هر Queue اولویت داریم. در شبیه سازی انجام شده ما تمام پردازش ها را به دو Queue مختلف تقسیم کرده بنا بر اولویت Kernel آنها، یعنی اینکه آیا از طرف سیستم هستند یا از طرف کاربر (System/User). بعد از تقسیم بندی و دسته بندی پردازش ها، هر Queue را با توجه با الگوریتم خاصی مرتب میکنیم. برای این شبیه سازی از الگوریتم Shortest Job First برای پردازش های سیستمی (Kernel) و از الگوریتم Round Robin با کوانتوم 1 برای پردازش های کاربر (User) استفاده شده. سپس باید پردازش ها را به ترتیب اجرا کنیم که در اینجا ما صرفاً Print کرده ایم. به عبارتی Print کردن پردازش ها معادل اجرا شدن و خارج شدن آنها از هر صف است.

میتوانستیم از الگوریتم های گوناگونی برای اینکار استفاده کنیم و یا تعداد صف های بیشتری داشته باشیم اما ساختار تغییری نمیکرد و ثابت میماند.

با ساختن یک Struct از نوع Process به متغیر های Internal یا داخل هر پردازنده مقدار دهی کردیم.

در حالت کلی این الگوریتم پردازش ها بر اساس اولویت، CPU time، IO time، IO access و سبب مموری و ... تقسیم بندی میشوند و سپس بر اساس اولویت بین Queue ها هم دوباره سورت میشوند اما اینجا برای راحتی کار و صرفاً نشان دادن ساختار و نه پیچیدگی های موجود، صرفاً دو گروه و بر اساس Mode آنها تقسیم بندی انجام شده است.

اولویت داشتن خود صف ها هم از اهمیت بالایی برخوردار است به طوری که به طور معمول صف اختصاص یافته به پردازش های CPU باید سریع تر از IO اجرا شوند چرا که زمان کمتری برای اجرا شدن نیاز دارند. یا پردازش هایی که برای Kernel هستند به طبع اولویت بالاتری باید داشته باشند نسبت به پردازش های کاربر. پس اولویت دادن یا سورت کردن خود Queue ها را هم باید در نظر بگیریم.

برای مشاهده نتایج نظیر میانگین Turnaround time و Waiting time تابعی تعریف کردیم و برای الگوریتم های مختلف (بعد از اجرای الگوریتم های مختلف بر روی آرایه مورد نظر) آن را صدا میزنیم. این کار را 3 بار انجام میدهیم، یک بار برای صف سیستم، یک بار برای صف کاربر و بار دیگر برای مجموع این دو تا ارزیابی نهایی انجام دهیم.

خروجی الگوریتم به ازای 3 پردازنده، 2 تا برای سیستم و 1 برای کاربر:

```

175
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

Enter number of processes: 3
Enter the Burst Time of Process 0:1
System/User Process (0/1) for Process 0?0
Enter the Burst Time of Process 1:2
System/User Process (0/1) for Process 1?1
Enter the Burst Time of Process 2:3
System/User Process (0/1) for Process 2?0
2System Processes:

P_ID    Burst    Waiting Turnaround
0        1        0         1
2        3        1         4
Avg Waiting-Time: 0.500000      Avg Turnaround-Time: 2.500000

User Processes:
[1] Process is running...

P_ID    Burst    Waiting Turnaround
1        2        0         2
Avg Waiting-Time: 0.000000      Avg Turnaround-Time: 2.000000

All:

P_ID    Burst    Waiting Turnaround
0        0        1         0
0        1        0         1
2        3        1         4
Avg Waiting-Time: 0.666667      Avg Turnaround-Time: 1.666667
keivanipchiagh@ubuntu:~/Desktop/AUT-CE-OS/Lab/Project$ 

```