

Jegyzőkönyv

1.feladat:

Round Robin nélkül:

Ebben az esetben a processzek 100 óraütésnél váltják egymást, így a 100. ütésig A, 200.ütésig B és a 201.ütésnél a C jelű processz kezdi meg a futását.

Ekkor a 100. ütésig növeljük p_cpu értékét, majd újra számoljuk a p_cpu és a p_uspri értékét a megadott képletek alapján:

$$p_cpu = p_cpu * KF, \text{ ahol } KF \text{ értéke } 1/2;$$

$$p_uspri = P_USER + p_cpu / 4 + 2 * p_nice;$$

Round Robinnal:

A Round Robin esetében minden 10. óraütésnél váltani kell a processzek között a 100. óraütésig.

Minden 100. ütésnél ki kell számolni a korrekciós faktort a futásra kész folyamatok alapján, újra kell számolni először a p_cpu értékét majd a p_uspri értékét. A futás azzal a processzel folytatódik amelyiknek a legmagasabb a prioritása.

Ebben a példában a 100. óra ütésnél a C jelű processznek lesz a legmagasabb a prioritása, ezért a 200. óraütésig ez a processz fog futni.

Majd ezután megismétlődik az újra számlálás, hasonlóan mint a 100. óraütésnél.

Ebben a példában a 200. óraütés után a D jelű processznek lesz a legnagyobb a prioritása, így a 300. óraütésig biztosan a D jelű processz fog futni.

A bővebb számítás megtalálható a Round_Robinnal.xlsx fájlban a 100. és 200. óraütésnél.

2.feladat: A magyarázat megtalálható a c állományban.

Források:

RR_nelkul.xlsx

Round_Robinnal.xlsx

AXF9AW_open_close:

main.c

AXF9AW.txt