

Operációs rendszerek

12.Gyakorlat

Memóriamenedzselés - Lapcsere algoritmusok

(GitHub repo-ban megtalálható Excel táblázatban vannak a számítások)

Czobor Patrik

KLYSO1

1. „Adott egy igény szerinti lapozást használó rendszerben a következő laphivatkozás és 3/4

fizikai memóriakeret a processzek számára.

Laphivatkozások sorrendje: 7 6 5 4 6 7 3 2 6 7 6 5 1 2 5 6 7 6 5 2

Memóriakeret (igényelt lapok): 3/4 memóriakeret.

Mennyi laphiba keletkezik (három és négy memóriakeret esetén) az alábbi algoritmusok

esetén: FIFO, LRU és SC? Hasonlítsa össze és magyarázza az eredményeket.

FIFO esetén: First In First Out algoritmust használva 3 lapnál 16, 4 lapnál pedig 14 laphiba keletkezik.

Ez egy nagyon alap algoritmus, hiszen csak azt figyeli, hogy szerepel-e a lapokban az érték, és ha nem akkor beírja a legrégebben hozzáadott helyre az értéket. Ha szerepel akkor meg laphiba nélkül tovább megy.

LRU esetén: Least Recently Used algoritmust használva 3 lap esetén 15, 4 lap esetén pedig 10 az összes laphibák száma.

Ennél az algoritmusnál nem a legrégebben hozzáadott értéket „cseréli le”, hanem a legrégebben nem hivatkozott értéket, így a gyakran használt értékeket tovább bent tartja a lapokban.

SC esetén: Second Chance FIFO algoritmust használva 3 lap esetén 16, 4 lap esetén pedig 12 az összes laphibák száma.

Ez a FIFO algoritmusnak egy bonyolított verziója. Itt minden lapokban szereplő érték kap egy mutatót, ami lehet 0 és 1. Ha kapunk egy elemet, ami benne van egyik lapunkban, akkor annak az elem értékéhez fűzünk egy 1-es mutatót. Ha kapunk egy elemet, ami nincs benne a lapokban, akkor mint a FIFO-nál a legrégebben hozzáadott helyre írja az új értéket, de csak akkor ha a mutatója 0. Ha a mutatója a cserélni kívánt értéknek 1, akkor lenullázzuk és a soron következő leg régebbi értéket vizsgáljuk.