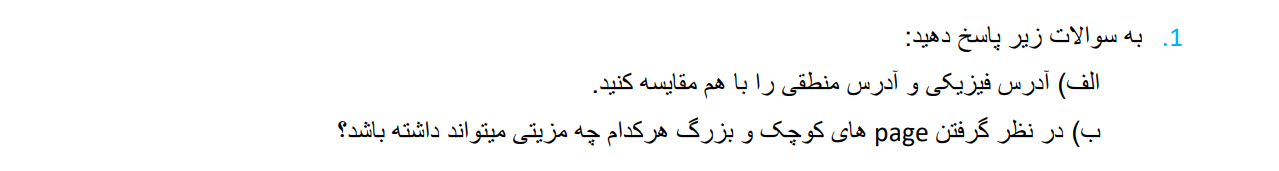
Amirali Farazmand

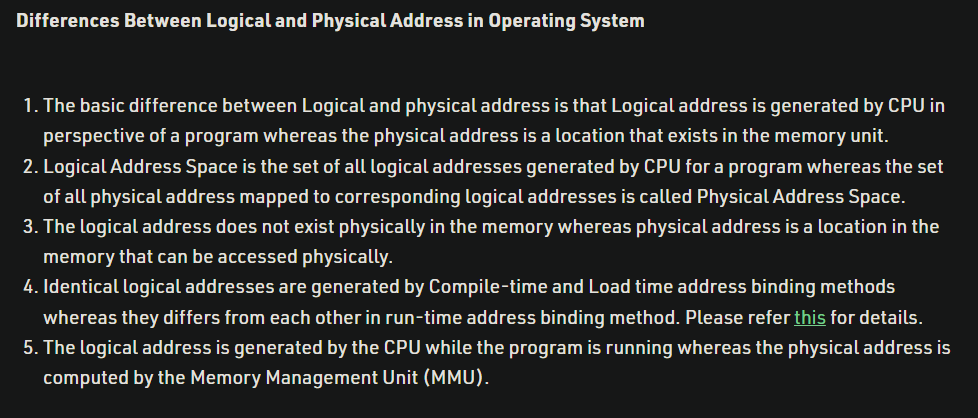
99522329

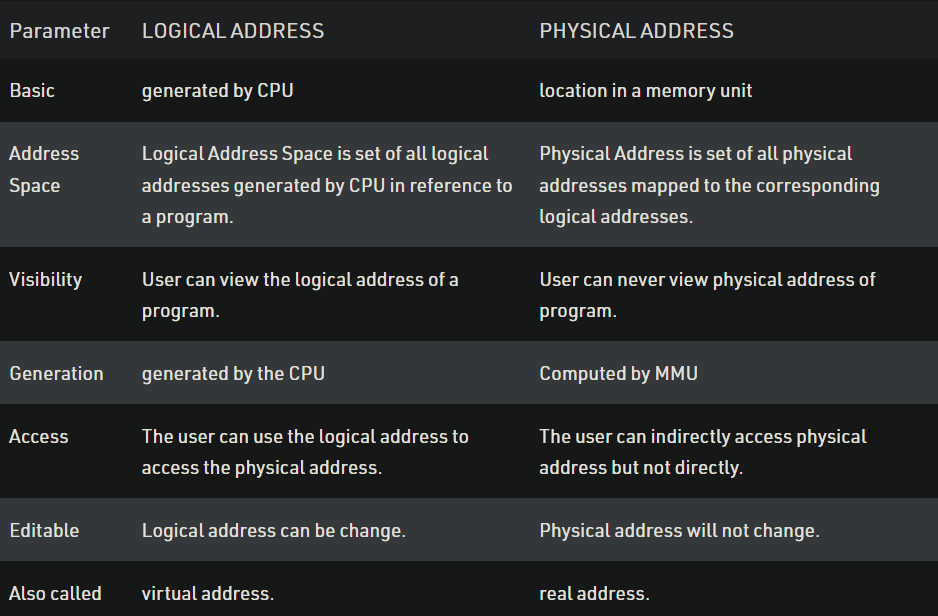
HW5



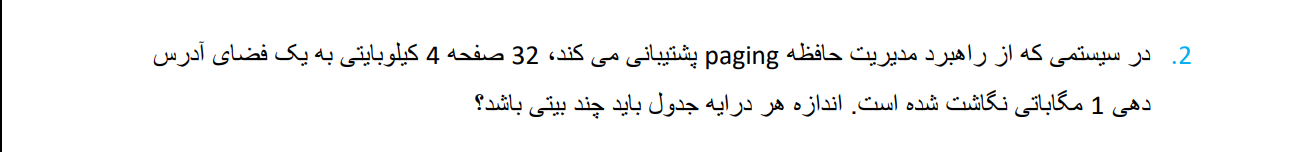
الف) آدرس فیزیکی آدرسی است در مموری واقعا وجود دارد، یوزر با physical address به طور مستقیم سروکار ندارد ، logical address (virtual address) آدرسی است که از طرف CPU تولید میشود و طی فرایندی ،توسط memory-management-unit(MMU) به آدرس فیزیکی مپ میشود.

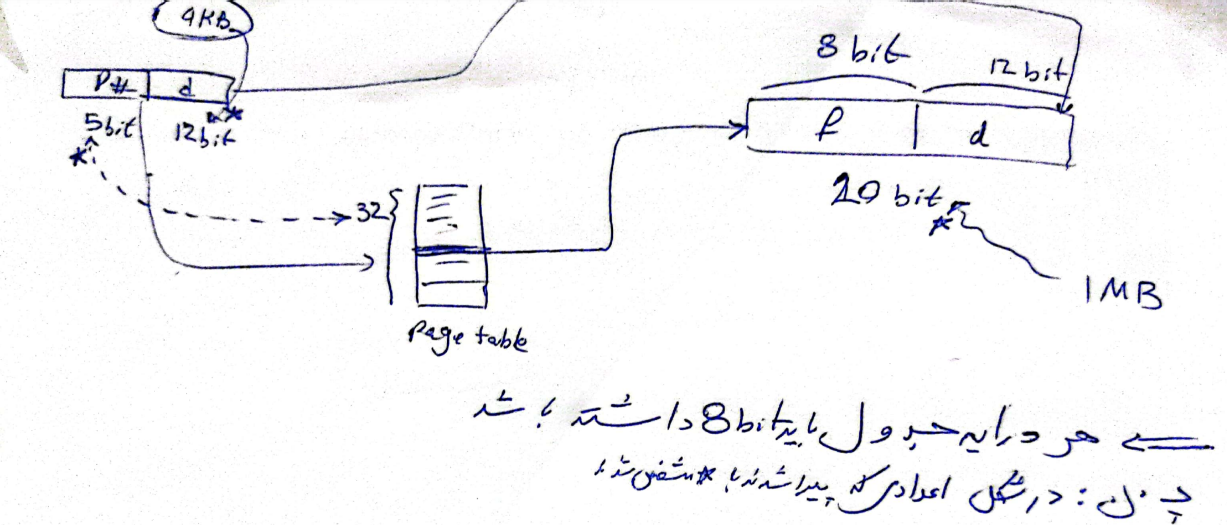
[مقایسه ی کامل تر و جواب کامل به سوال(GfG):](https://www.geeksforgeeks.org/logical-and-physical-address-in-operating-system/)

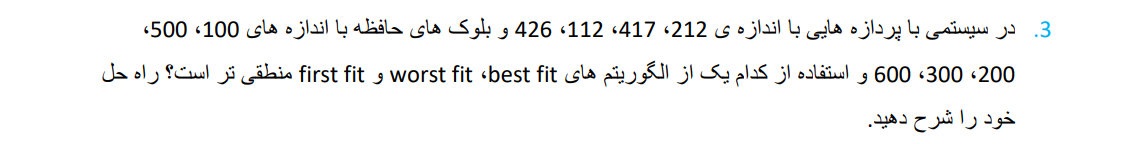


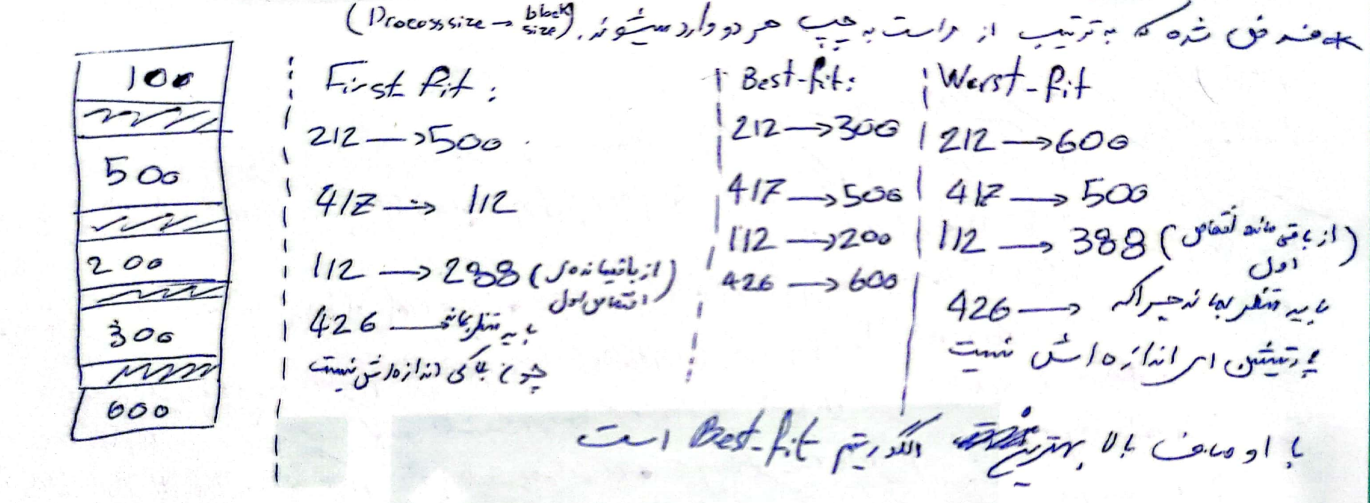


ب) اگر پیج ما خیلی کوچک باشد،page-table ما خیلی بزرگ میشود و بخش زیادی از مموری را میگیرد.و overhead زیادی از نظر زمان زیادی که در مموری است میدهد. در صورتی که اگر خیلی هم بزرگ باشد فضاهای خالی داخل مموری زیاد میشود و نمیشود بهینه از آن استفاده کرد.

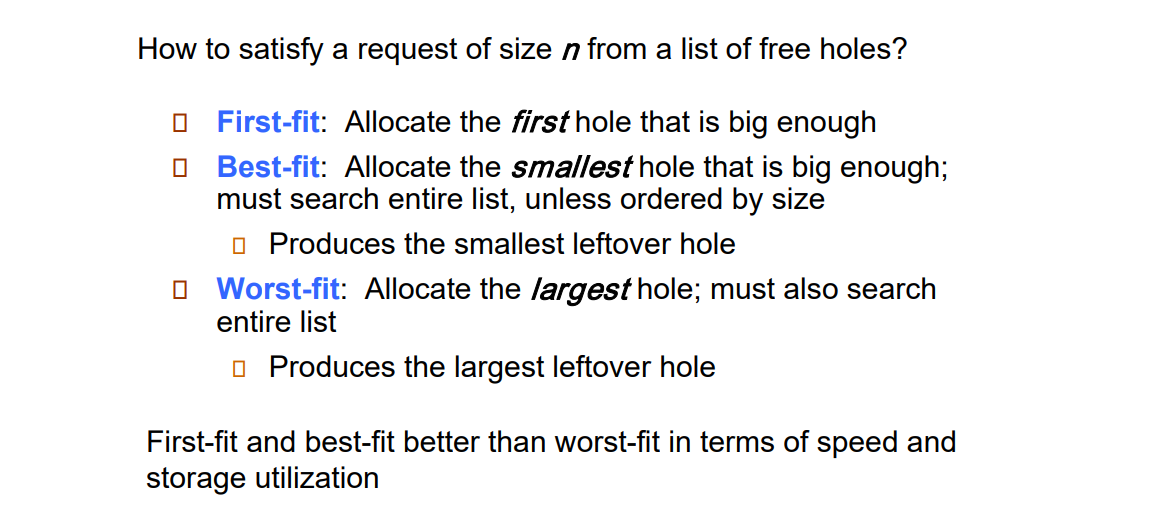


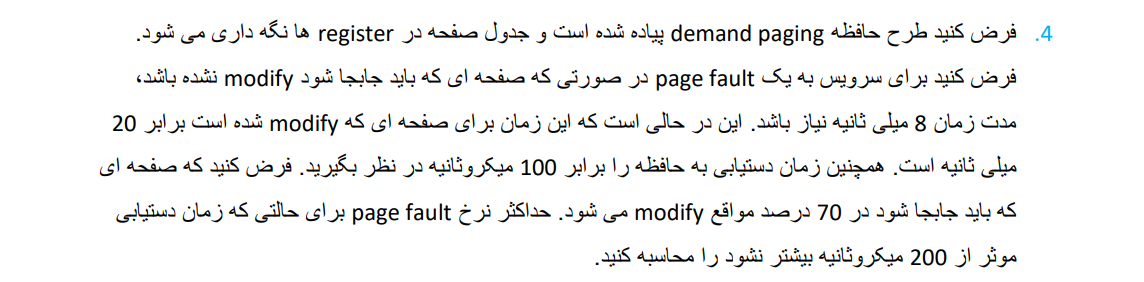


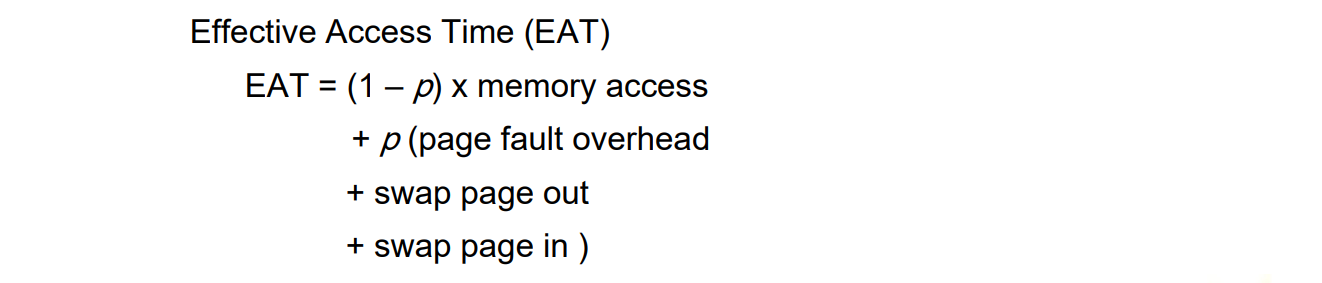




همانطور که دیده شد best-fit بهترین الگوریتم است. چرا که همه پراسس ها در مموری قرار گرفتند بعلاوه ی اینکه در این الگوریتم فضاهای خالی کوچکتری پس از allocate بوجود می ایند.

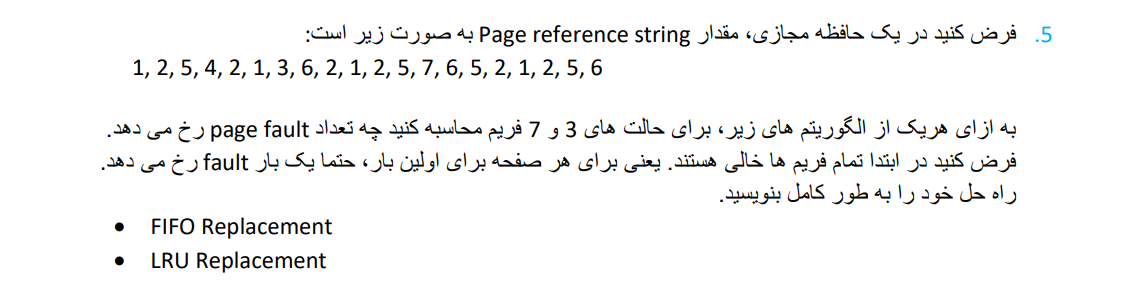


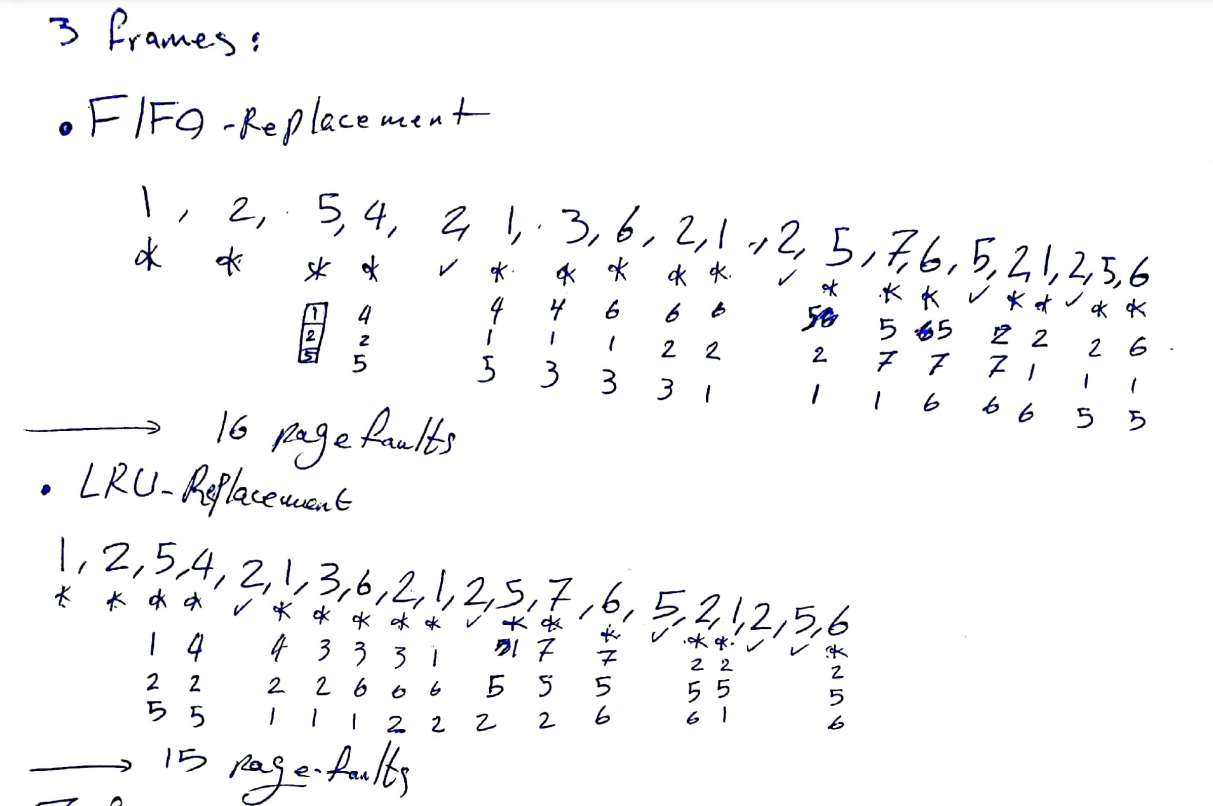


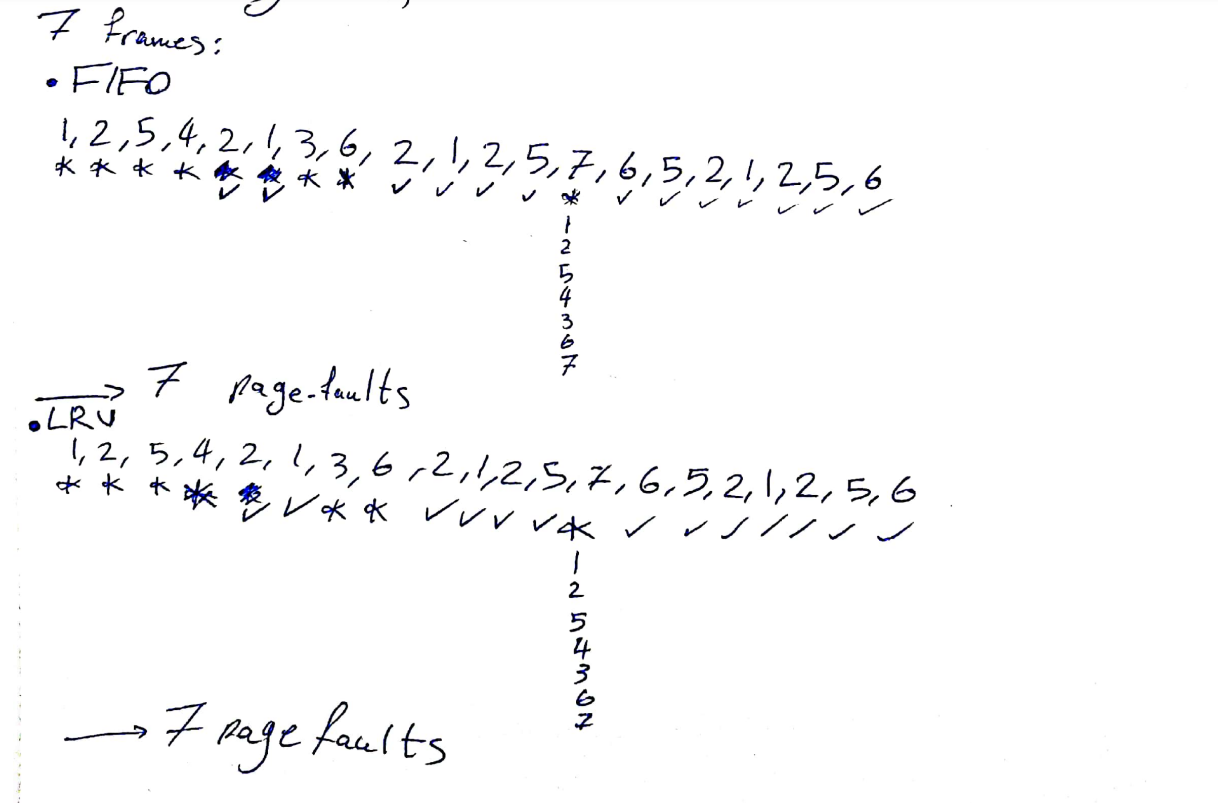


(1-p) \* 100 + p \* (0.7 \* 20 000 + 0.3 \* 8 000) <= 200

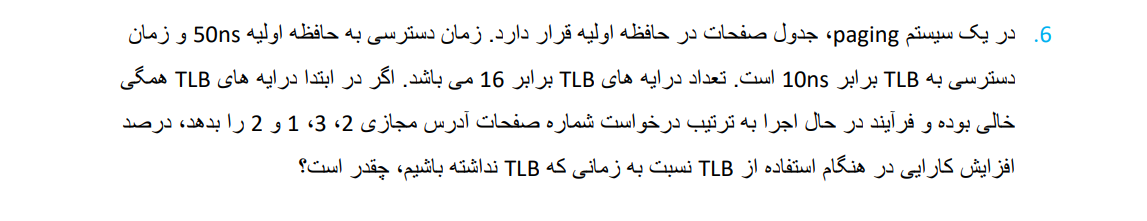
P<=0.006134969325







زیر جاهایی که page-fault میخورد با (\*) مشخص شده و همچنین page-fault های در مراحل اول چون صرفا فریم های خالی پر میشوند نشان داده نشده اند.



Without TLB:

Each process needs 50 + 50ns

With TLB:

TLB hits -> 10 + 50ns

TLB miss -> 10 + 50 +50ns

With TLB we have 3 misses and 1 hit, so:

Time = 3\*110 + 1\*60 = 390ns

Without TLB we have:

Time = 4\* 100 = 400ns

400/390 = 1.0256

2.56% = افزایش درصد کارایی