

Bonus

1.

記憶體優化上，demand paging 可以很 greedy 的保證有用的才在 memory 裡面，暫時沒用的在 disk 裡面，所以記憶體的使用量變得很少，優化得很好。

External fragmentation 方面，demand paging 不會有這方面的問題。因為任何空出來的 frame 都可以是 free frame 可以被拿來使用。

相對於其他 paging 方式而言，I/O 數量會變得比較少，因為是沒事不拿有事才拿得 greedy 方法，所以會減少 I/O 的數量。

Physical memory 方面，因為資料沒事就在 disk 上，所以不需要太多的 Physical memory。也不會因此受到限制。

Demand paging 可以更好的 share memory，因為不需要把共用的資料複製到每個需要得 process，只要放在 page 裡，不同 va 指到同一個 pa 就可以共用了。

但因為一開始所有資料都在 disk 上，所以相對於資料在 memory 上，interrupt 會變很多，因為很多 page fault。Data 的 access time 會因為需要一直 access disk，所以時間會拉很長。

2.

在 Multi-level Paging 中，每 1 個 level 都要 access 一次

memory，所以公式變成 EMAT (effective memory access time) =

$P \times (\text{TLB access time} + \text{memory access time}) + (1-P) \times (\text{TLB access time} + (k+1) \times \text{memory access time})$

如果 $k=3$ ，TLB access time=20ns， $P=0.8$ ，

$\text{EMAT} = 0.8 \times (20\text{ns} + x) + 0.2 \times (20\text{ns} + 4 \times x) < 180\text{ns} \rightarrow x < 100\text{ns}$ 。