* 產生hit ratio 的結果：
  + 先將預測為cached的request放入cache(放到滿)
  + 若存取到cache中的request，將該request放到MRU端
  + 然後過一段時間，針對late ,mean ,soon queue做demoting(位於soon的request就是直接踢掉)，但每個queue的MRU端前20% request除外，不做demoting
  + 接著開始從trace去讀request，這裡的request與上面的request是同一個trace
  + 然後看有多少request在cache內，多少存取的request不在
  + 最後計算hit ratio=

Eviction model

1. Request->cache model
2. If cached, then same request->duration model(predict duration label-🡪分別放入soon,mean,late list—當成queue使用 )
   * 一旦有request存取到這些queue中的data page，就將該data page放到queue前端(MRU端)
   * 這些queue，過一段時間後🡪access 5\*cache size request……不論這些request是否有被cache，就執行demoting
     1. 將所有位於late queue的data page放入mean, mean放入soon, soon…踢掉
3. 從soon的LRU往MRU找，找到就當作victim踢掉
4. Soon找完沒找到對象，接著從mean找，然後是late
5. 所以其實不需要eviction model。