

$O(1)$ nem sempre é igual a $O(1)$

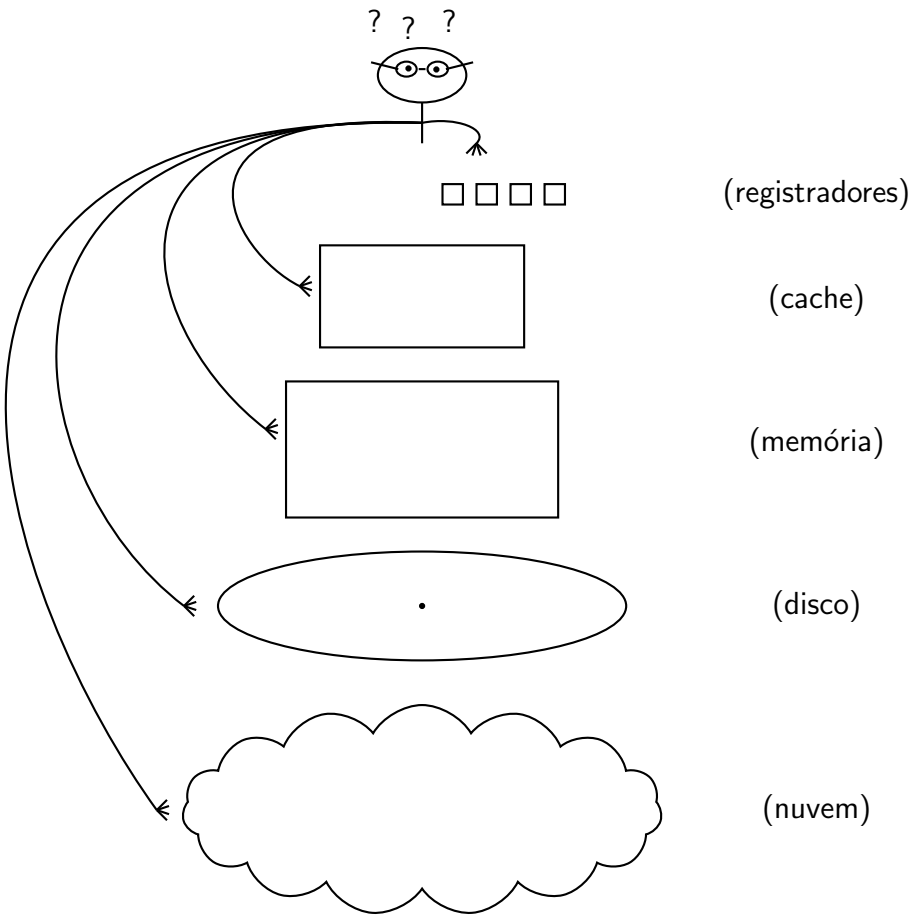
O resultado que nós obtivemos na aula passada foi muito bom mesmo

- A tabela de dispersão com encadeamento externo é uma estrutura de dados onde
- no pior caso, a busca leva tempo $O(\ln n)$
 - mas quase todos os elementos (99,9%) podem ser encontrados em tempo $O(1)$

Só que a gente ainda pode fazer melhor do que isso.

Quer dizer, a ideia de hoje é que um $O(1)$ nem sempre é igual a um $O(1)$.

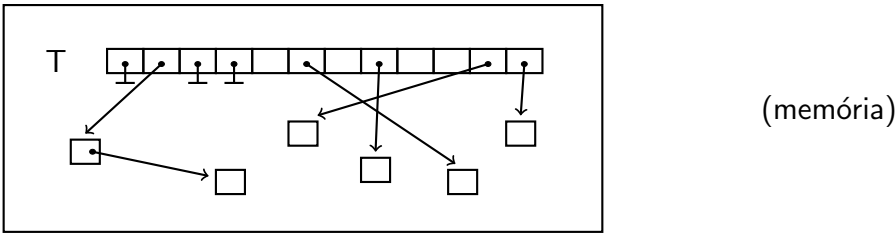
E a razão para isso tem a ver com a arquitetura dos nossos computadores



Quer dizer, uma operação $O(1)$ envolvendo dados que estão nos registradores ou no cache não é a mesma coisa que uma operação $O(1)$ envolvendo dados que estão na memória principal.

E quanto mais longe a gente vai, pior as coisas acabam ficando — (*apesar de tudo continuar levando tempo $O(1)$*)

O problema específico do encadeamento externo é que os registros que formam as listas encadeadas podem estar em qualquer lugar na memória



Daí que, no momento em que a gente percorre uma lista encadeada no procedimento de busca, a gente nunca espera encontrar o próximo elemento no cache.

Cada elemento inspecionado na busca envolve um acesso à memória principal.

E isso leva tempo pra caramba — (*apesar de ainda ser $O(1)$...*)