

! ierarhizarea memoriei

! un program poate fi rulat mai eficient dacă paralelizăm părți ale sale
avem 5 procesoare
p% din program poate beneficia teoretic de paralelizare / îmbunătățire
avem:

legea lui Amdahl | speed-up SI

$$S = \frac{1}{1-p + \frac{p}{S}}$$

legea lui Gustafson | speed-up SI

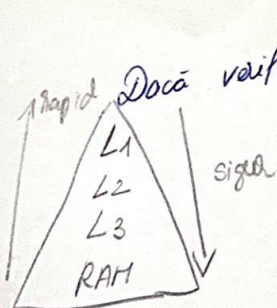
$$S = 1-p + Sp$$

1 Kbyte = 1000 bytes
1 MB = 1000 Kbytes
1 GB = 1000 MB
1 T = 1000 GB

IERARHIZAREA MEMORIEI:

registrii : acces imediat 100-1000 bytes
cache L0: acces foarte rapid 5-20 Kbytes
cache L1: 700 GB/s, 100-500 Kbytes
cache L2: 200 GB/s, 500-1000 Kbytes
cache L3: 100 GB/s, 1-5 MB
mem. principală = RAM, 100-500 MB/s, 2-6 GB
disc H/D SSD: 10-100 MB/s, 1 TB

De ce e bine să existe cache : mem. principală = 50ms
L1 = 10% șa nu găsim = miss rate
1ms
L2 = 5ms cu 1% miss rate
L3 = 10ms cu 0.2% miss rate



Doacă verificăm în RAM = 50ms

$$L1 = 1ms + \frac{1}{10} \cdot 50 = 6ms$$

$$L2 = 1ms + \left(\frac{1}{10} \cdot \left(5ms + \frac{1}{10} \cdot 50 \right) \right)$$

$$= 1ms + \frac{1}{10} \cdot (5ms + \dots) = 1.55ms$$

$$L3 = 1ms + \frac{1}{10} \cdot \left(5ms + \frac{1}{100} \cdot \left(10ms + \frac{2}{1000} \cdot 50ms \right) \right)$$

$$= 1.5101ms$$

Principiul de localizare:

i/o - input/output

RAM:

$a_0 \ a_1 \ \dots \ a_{999}$

L_1 :

a_0	a_1	a_2	a_3
-------	-------	-------	-------



L_1 :

a_4	a_5	a_6	a_7
-------	-------	-------	-------

L_2 :

a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

căutăm a_0 și nu e în L_1 , nici L_2 → mergem în RAM după adresă segmente din vord din RAM în L_1, L_2

căutăm a_1 și găsim în L_1

căutăm a_2 și găsim în L_1

— " — a_3 — " —

— " — a_4 nu e în L_1 dar e în L_2 deci actualizăm L_1

— " — a_5 și găsim în L_1

miss rate pt. $L_1 = \frac{1}{4} = 0.25$ sau 25%

miss rate pt. $L_2 = \frac{1}{10} = 10\%$

Localiza CACHÉ VS RAM

1) la citire nu sunt pb.

2) la scriere se complică: rez. este scris în cache și celelalte mem. tb. actualizate
 $L_1, L_2, L_3 \text{ RAM} \dots$