

## Měření daty výkonu souboru proktonovajícího časem

čas měří na mnoha faktorech (např. počet můžetelech s systémem, rozdílní ráz, pevnáni procesu, časovní diskryzace; přístupek do cache, ...)

jeden a tentýž program na komplex počítací může být různě rychlý  
jednou dobou

při měření času vyhodnocuje kvalitou hardware i operacemi systému  
tisk se experimentálně porovnává na různé konfigurace počítací  
(objekt s rozdílnou množinou systémových a uživatelských služeb)  
a v různém prostředí

dále některé postupy měření času nejsou přístupné nebo mame  
normálnímu aplikativnímu programování

v prostředí UNIX (LINUX) se používají 2 rážadlové přístupy -  
jeden je měření na časovaci s mimoúplavnou, který pracuje  
přesným procesorem - jde hardvérem přesně se časem, interval může procesu,  
druhý na čítačce <sup>clock</sup> taktu, který je mnohem abbivní  
jednotlivým cyklem - speciální registr, který se mikromonitory jde hardvérem  
první je přístup v rámci funkce `gettimeofday()`,  
která vrátí aktuální čas proktonování daným procesem  
nebo funkce `time()` v shellu, který vrátí čas proktonování  
uživatelského procesu a systémový čas  
rychleji jsou publikována a lze je použít pro měření delších programů  
(1s a delší)

druhý přístup se lze dát po mimo rážadlové programy, na některých  
systémech je přístup v rámci funkce `getitimer()`, na  
jiných se musí doprogramovat v assembleru, některé procesory  
čítačku taktu mohou měřit

rehozi se po delší programy, protože nezvlášť mohou jednotlivé procesory (procesor byl procesorem mnoha různých a koncem očekávaní vzdálí, například se mu někdo, co když měnil) - rozhodně možnost je v rozdílu

mezi cache - poskytuje rychlosti, protože programům má jen několik malých kontejnerů nad tím, kde je všechno a kdežto data jsou v cache a co se přesné dělá, když mají zjistit mnohem rychleji  
v poslední době se to hodně studuje a hodně se s tím experimentuje, ale výsledky jsou poněkud nejasné

existují 2 různé algoritmy - markovské algoritmy na mimo konkrétní architekturu nebo magické algoritmy, které jsou "soběstojící"

## mezi optimizátory

Nové problémy:

procesory méně preferují podle rychlosti

vícejednové procesory

parallelismus

Obecně je problematika přesného měření času složitá, metody se řídí méně spolu s vývojem hardwaru a operačních systémů.

Je tedy nutné vždy důkladně prostudovat parametry a možnosti konkrétního výpočetního prostředku.

Jednoduché doporučení:

odpojit počítac od sítě

nechat být jen sledovanou ulotou

omezit systémové služby na minimum

experimenty nelibidně provádět se stejnými daty