

IB002 - Základy informačních technologií

Asistentka: pazurikova@mail.muni.cz

Midterm exam - ve formě zkoušky

koncová známka: 0.2 mid + 0.8 zk = výsl. zn.

9/18/2012 - 01

Historie

Babbageův počítačový stroj

první programovatelný na základě instrukcí

programy psala sestra lorda Byrona (Augusta Ada King)

popsala jej a díky tomu 1991 rekonstruován a prokázána funkčnost

měl paměť

měl oddělenou datovou a programovou paměť

oddělená vstupní jednotka pro data a program

výstup programu

řídící jednotka uměla IF - GOTO (podmíněný skok)

Turingův stroj

teoretický výpočetní stroj - logická konstrukce

přechodový graf, množina počátečních a koncových stavů, páska na které je zapsána zpráva, podle symbolů přechází z jednoho stavu do druhého, důležité je, když dojde do jednoho z koncových stavů, podle toho jsme schopni rozhodnout, jak dlouho se bude nějaký problém počítat / zda má řešení

(Turing pracoval

na Enigmě

na řešitelnosti logických problémů

)

Von Neumanovo schéma

40. léta 20. stol.

dvojková soustava pro počítání

znát schéma (obrázek)

[obrázek=neuman_scheme]

sekvenční schéma - nelze dělat paralelisaci, mnozí říkají, že to informatiku zpomalilo

Technické prostředky

ALU - základní sekvenční jednotka

procesor, zařízení vykonávající instrukce, instrukční cyklus (vybrání a provedení instrukce)
lze skákat - výběr instr.
vnitřní hodiny - takt procesoru
pokud je jich více -> vnitřní paralelismus (dnes hlavně GPU)

Typy procesorů

Universální

CISC: kompletní instrukční sada
všechny instrukce, ale problém, že ne na všechny instrukce je potřeba stejně času
RISC: redukováná instrukční sada
komplikované instrukce má rozdělené na jednodušší subinstrukce

Specialisované

Embedded - vestavěné, velmi specifické využití, zadaný program - žehličky, auta, ...
Vector - masivně paralelní stroje - hodně ALU jednotek a zpracovávají úlohy paralelně
použití - vědeckotechnické výpočty, např. meteorologie

Paměť

slouží k uchování dat
deska sloupců a řádek
Přímá adresace - (sloupec × řádek)
cyklus - doba potřebná pro vystavení a zápis
vzpamatování po provedené operaci - také sežere nějaký čas
statická × dynamická paměť
Hierarchie pamětí - rychlost, cena, ekonomický imperativ v IT - prostředky závisí na penězích, v poslední době důležitý -- energetická náročnost

Periferie

vstup / výstup informací
základní: komunikace s uživatelem
mohou to být různé sofist. zařízení pro virt. realitu (brýle a helmy pro prostorvé vnímání, detekce pohybu, wearable computers)
s jinými systémy
počítač. síť, speciální sběrnice
permanentní ukládání dat

Permanentní ukládání dat - permanence je velmi relativní

Paměti jsou rychle, drahé, krátkodobé
Disky - pomalejší, levnější
Pásy - pomalý přístup, velmi levný a úsporný
Sítě - data kolují optickou sítí (200 km)

Sítě

drátové - metalické (elektrické signály) / optické (optický laserový signál)

bezdrátové
 radiové
 optické - IR světlo
 akustické
mechanické
 ???

Počítač

logický přístroj s klasickými komponentami, pomáhají řešit problémy rozložené na výpočty
neuronové počítače - model lidského mozku, spojení vznikají a zanikají, training
buněčné automaty

Paralelní systémy

pro případy, kdy nestačí jedna výpočetní jednotka - více jednotek, které spolu komunikují

Typy:

úzce propojené

společná paměť - nutno řešit, aby si nepřepisovaly procesory svá data
minimální vzdálenost procesorů - komunikace neovlivňuje výpoč. čas
speciální sběrnice
velmi jemný paralelismus - paralelně se provádějí instrukce, nikoliv celé programy
díky společné paměti nutno ji mít dostatečně velkou, komplikovaný rozklad úloh, ne všechny úlohy jsou vhodné
výpočetní model: sdílená paměť

volně propojené

většinou distribuovaná, vzdálenost procesorů už hraje roli - latence, speciální propojení
lze používat vzdálenou paměť
výpočetní model: zasílání zpráv - provádějí se větší kusy kódu, sdílí se výsledky

distribuované

vždy distribuovaná paměť, vzdálený přístup do paměti nelze
latence vysoká, obvykle přes LAN
výpočetní model: zasílání zpráv

gridy

geograficky rozsáhlé, popojují se samostatné počítače
WAN Internet
extrémní latence - zasílání zpráv
zpracování dat např. v LHC (nejde moc o rychlost zpracování)
sít výpočetního výkonu

Číselné soustavy

dvojková, šestnáctková, osmiková, dvojková
nejpřirozenější je desítková

Desítková

$$819 = 8 \times 10^2 + 1 \times 10^1 + 9 \times 10^0$$

první počítače fungovaly na desítkové, ale nebylo to moc účinné a realizovatelné

Dvojková

$$\begin{aligned} 819 &= 1 \cdot 2^9 + 1 \cdot 2^8 + 0 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 \\ &\quad \quad \quad 512 \quad \quad \quad 256 \quad 128 \\ &= 1100110011 \end{aligned}$$

Octa & hexa

jednoduše převoditelná z binární, podobně hexa

Rozlišitelnost

neexistuje nekonečný desetinný rozvoj
nejdelší desetinný rozvoj, který jsme schopni zapsat

$$X + \text{eps} > X$$

$$X + (\text{eps}/2) = X$$

Zápis reálných čísel

mantisa

$$0 \leq m \leq 1 \text{ AND } x = m \cdot 2^e$$

Záporná čísla

- přímý kód - nula/jednička na začátku
- inverzní kód - binární negace
- dvojkový doplňkový kód (binární negace + 1) - nejčastější

Přímý kód

- přidává se znaménkový bit
- --> máme dvě nuly

Rozsah: <-MAX, -0> a <+0, +MAX>

Inverzní kód

- opět jsou dvě nuly

Rozsah: $\langle -MAX, -0 \rangle$ a $\langle +0, +MAX \rangle$

Dvojkový doplň. kód

- vychází z inverzního

- má jen jednu nulu, ale asymetrický rozsah

Rozsah: $\langle -MAX, -1 \rangle$ a $\langle 0, +MAX \rangle$

Rozsahy čísel

Podle počtu bitů

8 bitů = 1 Byte

Racionální čísla

- podle normy IEEE 754

- součásti

- znaménko

- mantisa (přímý kód)

- exponent (kód posunutě nuly)

Normalisace mantisy

- zapamatovat si, že mantisa je normalisována (nezapisovaná jednička)

Přesnost (rozdíl mezi dvěma po sobě jdoucími dílky): počet bitů mantisy + 1

Operační systémy

OS se stará o to, co je pro všechny programy společné - zavádění programů, ukončování programů, etc.

Bootstrap loader

- smyčka, v níž OS běží a předává řízení jednotlivým procesům

Po něm Spooling - program je nezávisle zaveden na vykonání

- možno zavést prg, čekal ve frontě (pool) a spooler vybíral prg podle priority

- zaveden Direct Memory Access a multiprogramování

- zpracovávání dávek (neexistuje přímý přístup)

Timesharing

- všechny zavedené úlohy se přidělují po časových kvantech (virtuálně se počítají současně)

- umožnil zpracování interaktivních úloh a zavedení disků

- 60. léta (začátek)

Úkolem OS je zjednodušení práce, odbourání nutnosti práce s nejnižší úrovní počítače

Zajišťuje také sdílení vzácných zdrojů - postarat se o funkčnost sdílení, jeho účinnost, bezchybovost a bezpečnost přístupu (autorizace / autentizace)

AAA - autentizace, autorizace, accounty

co jsem co smím kolik toho použiju
(authentication)

Problém časování

Ne všechny součásti jsou stejně rychlé, řádové rozdíly

(za dobu přečtení 1 Bytu z HDD může CPU udělat cca 10 milionů operací)

--> prokládání I/O operací a výpočtů

těžkopádně a ne vždy účinně

proto se procesoru přiděluje více procesů, prokládání operací nemusí být v rámci jednoho procesu

Vybírání procesů:

Priority:

- statická
- dynamická

Paměť

- OS udržuje v paměti pouze ty procesy, které jsou právě aktivní - virtualisace paměti

- každý proces si tedy myslí, že má celou paměť

- musí se zavést ochrana paměti, aby jeden program nepřepisoval data jiného programu

Základní složky OS

[slide přednášky]

S rozvojem OS se OS zvětšovaly - monolitické operační systémy

Velmi těžké opravovat nějaké chyby, téměř nebylo možné vyměnit HW pod monolitickým OS

--> Úprava OS - rozdělení na několik částí

- kernel (jádro) a další součásti

Procesy

Programový čítač v CPU ukládá hodnotu, kde CPU skončil u které hodnoty

02.10.2012 - 3. přednáška

Paměť

- dvě základní operace - **alokace**, **dealokace**

- doplňková operace - **realokace** - změna rozsahu paměti

- **čištění paměť** - garbage collection - použito v případě, že proces skončí chybou a už nedealokuje svou paměť

- **virtuální paměť** - nepoužitá paměť se přesouvá na HDD, aby se optimálně využila drahá operační paměť

- **swapping** - při nedostatku paměti se celé procesy odsouvají na disk, popř. se swapují i díry

- **paging** (stránkování) - rozdělení paměti na konstantně velké stránky, při alokaci nebo swappingu se pracuje s celými stránkami

- **segmentation** - stejné jako stránkování, ale segmenty nejsou konstantně velké → ušetření paměti

Ochrana a bezpečnost

- může nastat přístup procesu do místa paměti, které mu nepatří
- znepřístupnění služby (denial server)
- troyan horse - proces, který se vydává za něco, co není
- viry - procesy vzniklé za účelem škodit
- operační systém musí kontrolovat procesy, protože v každém programu jsou chyby

Zásady bezpečných systémů

- zveřejnění algoritmů - spousta lidí, kteří mohou dané algoritmy kontrolovat
- standardní nastavení je zákaz přístupu - povoluje se pouze tam, kde to je potřeba

Client-server model

- server a řada klientů
- jeden z modelů distribuovaného počítání, asymetrické rozložení - klient posílá požadavku a server je vybavuje, klienti spolu nekomunikují
- klient a server jsou samostatné procesy
- druhý model je [peer2peer](#) - není server, klienti spolu komunikují jako rovný s rovným

Společné vlastnosti komunikačních systémů

- příklady: telnet, ssh, WWW, X-Windows systém na Unixu
- ???

Komunikační vrstvy:

- třívrstvý model - datová vrstva, logická vrstva, presentační vrstva
- klienty jsou rozlišovány na fat a thin
 - fat toho umí hodně sám, hodně lokálních zdrojů, po serveru toho chce hodně málo (pouze data)
 - thin umí málo, většina operací je na serveru

Škálovatelnost - důležitá (co to je?)

Middleware

- zkratka v rámci protokolů - komunikaci probíhá na vyšší abstraktní úrovni
- příklad: přenos souborů, vzdálené připojení, ...

Mobilita

- návrat k tečným klientům, protože mobilní zařízení nebyla tak výkonná, ale opět se vyvíjejí zase tlustší klienty, protože výkon na straně smartphonů je vyšší
- inherentně distribuované
- client-server model

Principy návrhů systémů

- efektivita, mobilita, robustnost, přenositelnost, kompatibilita, flexibilita

Efektivita

- maximální využití zdrojů, jasné principy
- dekompozice návrhů - využití standardních technik, které už existují (obj. programování)

Robustnost

- schopnost reagovat na výpadek

- řešeno redundancí (inženýrské řešení - snižuje efektivitu)
- trojí jištění systému (pokaždé sadě operací se kontroluje výsledek, v případě neshody se bere za správný ten, na němž se shodli dva, případně se opakuje) - řešení u atomových ponorek USA
- v současné době zájem o self-healing programy

Bezpečnost

- omezuje útočníka, ale i uživatele - snižuje snadnost použití, klade nároky na správu systému

Klasifikace OS

monolitický, vrstvený, modulární, micro-kernel

Monolitický OS

- původní proprietární OS
- rozsáhlé, špatně udržitelné a rozšiřitelné systému
- použitelný, protože se HW téměř neobnovoval
- každému HW byl jeden OS na míru

Vrstvený OS

- vrstvy odpovídající architektuře - správa CPU, RAM, periférií, FS
- v případě změny jedné komponenty šlo změnit jednu vrstvu a zbytek ponechat

Modulární OS

- moduly místo vrstev
- zhruba obj. programování
- nebezpečí vzniku zbytečného SW (fatware)

Kernel- OS

- jádro - složka pro alokaci správy zdrojů, přímý přístup k HW a bezpečnost
- *malé je pěkné*
- malé moduly, snadno přepisovatelné

API

- aplikační programové rozhraní
- definuje způsob přístupu k OS a službám
- říká jakým způsobem můžeme něco dělat (např. otevřít, číst, psát a zavřít program)
- programy volající služby API nemusí vůbec vědět, že se změnil nějaký HW

09.10.2012

Sítě

16.10.2012

1. počítačové sítě

protokoly neřízeného přístupu – při kolizi je utlumen jeden vysílající (slide 35/59)

~ řízeného přístupu: rezervace časových intervalů, ...

L2 vrstva – datový spoj

kolizní doména - na jednom přenosovém médiu, kolize pokud vysílá více stanic

identifikace signálu, aby daná cílová stanice poznala, že to je pro ni: adresace (na začátku signálu)

bus network – všechny stanice na jednom kabelu

token network – "pešek" – předávání tokenu, kdo má token, může mluvit

star topology – centrální hub, pro zničení sběrnice musí být zničen hub/switch/bridge, stanice připojeny samostatně

23.10.2012

Síťová vrstva

spojování více fyzických LAN sítí

tvorba packetů

fragmentace – pokud síť nepodporuje dostatečnou velikost packetů, rozdělí se na více menších

u IPv6 se velké packety zahazují a odesílá se ICMP zpráva se žádostí o nový packet se správnou velikostí

každé síťové zařízení musí mít jednoznačnou identifikaci – IP adresa

síťová vrstva zařizuje, aby se packety přibližovaly ke svému cíli

překlad IP a mac (fyzických) adres

internetworking – propojování sítí

internet – jakékoliv propojení více sítí

Internet – jedna konkrétní síť (celosvět. Internet)

IPv6 – adresy hexadec.

ICMP – internet control message protocol

6.11.2012

L4 – transportní vrstva

přidává k adrese port – aby se nepletla data různých aplikací na jedné IP adrese

na této vrstvě se buduje technologie spolehlivého přenosu – doručení všech dat a ve správném pořadí

L4 poskytuje služby pro aplikační vrstvu, se síťovou vrstvou zajišťuje doručení dat a spolehlivost přenosu, poskytuje jakýsi virtuální komunikační kanál → end-to-end služby. Protipól je na nespojované síti step-by-step systém – tohle udělá virtuální spojovanou síť

congestion control – kontrola zahlcení

Umět:

projit vrstvy a napsat, jaká je adresace a jaká je tam základní jednotka (e.g. na síťové vrstvě packety, adresy)

Multiplexing – zajišťuje, aby se data od různých app nepromíchala spolu

UDP

User Datagram Protocol

používáme jej, abychom propojili dvě app tak, aby spolu mohly komunikovat a data se nemíchala velmi jednoduchý, minimální režie

TCP spojení

zajišťuje o

positive acknowledgement – potvrzje se přijetí

negative acknowledgement – oznamuje se nepřijetí