

Számítógép architektúrák

Számítógépek, számítási modellek

Bemutakozom ...

Dr. Vadász Dénes, tanszékvezető egyetemi docens

vadasz@iit.uni-miskolc.hu

<http://www.iit.uni-miskolc.hu/~vadasz>

Informatikai Intézet épülete,

I. emelet, 109. szoba

Miskolci Egyetem

Gépészmérnöki és Informatikai Kar

Informatikai és villamosmérnöki tanszékcsoport

Általános informatikai tanszék

<http://www.iit.uni-miskolc.hu>

A tárgy címe, célja

- Számítógép (computer)
 - Egy *programozási nyelv* segítségével leírt *számítási feladat* végrehajtására szolgáló *eszköz*¹
- Architektúra (felépítés, szerkezet)²
 - Funkcionális specifikáció az irányultság: specifikáció,
 - Megvalósítási célú irányultság: egységek és kapcsolódásuk.
 - Bármelyik irányultságnál különböző részletezettség
- A tárgy célja: általános hardverismeret megszerzése, továbbá felhasználói felületek (parancsértelmezős és grafikus) megismerése.

Az oktatási módszereink ...

- Vetített képes előadások ...
 - <http://www.iit.uni-miskolc.hu/~vadasz/GEIAL301B>
 - itt az előadások képei nyomtatható formában (Acrobat Reader)
 - A Számítógépek, számítógép rendszerek c. jegyzet
- Az előadásokon a lényeges dolgok kiemelése, fontos definíciók, konvencionális szóhasználat stb.
- Laboratóriumi gyakorlatok
 - Kötelező a látogatásuk
- Önálló feladatok

GPRAK
Agyakorlatok

A teljesítés feltételei

- Alíírás és vizsga
- Az alíírás feltételei
 - A gyakorlatokon aktív jelenlét, az ottani feladatok eredményes elvégzése. Legalább 8 gyakorlatot el kell ismertetni! Köztük a „szerelés” kötelező!
 - Az évközi zárthelyi dolgozat eredményes megírása.
 - Alíírást csak a tanulmányi időszakban szerezhethetnek!
- A vizsga írásbeli és szóbeli vizsga
 - A „beugró írásbeli” után a szóbelin tételekről

Az ütemterv

- A tárgy honlapján, a tanszéki hirdetőtáblán ...
- Témák:
 - Számítógép történet, számítási modellek
 - A felhasználó szemlélete. Szolgáltatások.
 - Parancsnyelvi felhasználói felület: a burok (sh, bash)
 - A CPU
 - Memória
 - Sínek
 - Eszközök: képernyő, billentyűzet, mutatók
 - Háttértárak, nyomtatók

Számítógép történet

- Tanulmányozzák Katona István A számítógép története c. prezentációját!

http://www.ektf.hu/mediainf/inf/ktoth/konyvtar/Szamitogep-tortenet_elemei/frame.htm

- Ebből fontos: a Neumann elvű gép
- A Neumann elv röviden:
 - A számítógép legyen teljesen elektronikus, külön vezérlő- és végrehajtó egységgel.
 - Kettes számrendszert használjon.
 - Az adatok és a programok ugyanabban a belső tárban, a memóriában legyenek.
 - A számítógép legyen univerzális Turing-gép¹.

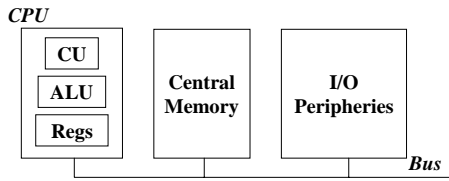
Az utóbbi évtizedek a számítástechnikában

Az évek	60-as	70-es	80-as	90-es
A paradigma	Kötegelt feldolgozás	Időosztás	Asztali gépek	Hálózatok
Hol?	Számító-központban	Terminál-szobában	Íróasztalon	Mobil
Az adatok	Numerikus adatok	Szövegek + számok	... + rajzok	Multimédia
Fő cél	Számítások	Hozzáférés	Megjelenítés	Kommunikáció

Az utóbbi évtizedek a számítástechnikában₂

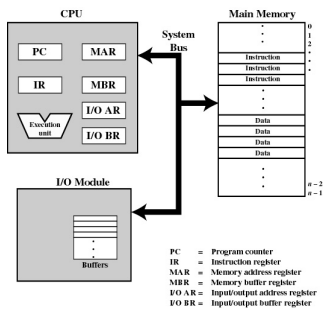
Az évek	60-as	70-es	80-as	90-es
A paradigma	Batch	Time sharing	Desktop	Network
Az interfész	Lyuk-kártya	Billentyűzet + CRT	Lásd és kattints	Kérdezd és mondd
Kapcsolódás	Nincs	Terminál vonalak	LAN	Internet
Tulajdonos	Intézeti sz.központ	Osztályok	Osztályok dolgozói	Mindenki

A Neumann architektúra



- Control Unit: Vezérlő egység
- ALU: Aritmetikai és logikai egység
- Regs: Regiszterek
- Central Memory: Központi memória, tár
- I/O Peripherals: Ki/bemeneti egységek, perifériák
- Bus: Sín, adattovábbító áramkörök
- Central Processing Unit: Központi feldolgozó egység, processzor

Egy másik funkcionális modell



- PC = Program counter
- IR = Instruction register
- MAR = Memory address register
- MBR = Memory buffer register
- I/O AR = Input/output address register
- I/O BR = Input/output buffer register

Figure 1.1 Computer Components: Top-Level View

A központi tár és perifériák

- A memória
 - Adatokat (bit, bájt, szó, blokk, mezőkből álló rekord, fájl stb.) és
 - gépi instrukciókat tartalmazó,
 - címezhető cellák (rekeszek) készlete.
- A perifériák (ki/bemeneti egységek)
 - Periféria vezérlő áramkörökkel (controller, adapter) kapcsolódó eszközök (devices).
- A CPU
 - A gépi instrukciókat feldolgozó (processzáló) egység, a processzor. Több funkcionális elemből áll (CU, ALU, Regs stb.)

A Neumann gép működése

- A memória rekeszeiben ott vannak a gépi instrukciók (a kód, a program) és az adatok.
- A CPU memóriából felhossa (fetch) a soron következő gépi instrukciót
- A CU elemzi az instrukciót. Értelmezi.
- Ha szükséges, a memóriából felhossa az instrukció operandusát
- Az ALU végrehajtja az instrukciót
- A végrehajtás eredménye a regiszterekbe, esetleg a memória megfelelő rekeszébe kerül
- Folytatódik a soron következő instrukcióval ...

Az állapotterek

- Vegyük észre e következő absztrakciókat
 - A „soron következő instrukció” koncepció egy *instrukció folyam* (instruction stream) képzetet ad
 - Ezen folyamon egy mutató mutathatja a soron következő elemet. Ez a mutató a programszámláló regiszter (PC: Program Counter, IP: Instruction Pointer)
 - Az instrukció folyam instrukcióinak készlete *vezérlési állapotteret* határoz meg. Ebből egy állapotot az ad meg, hogy az i-edik lépésben mely instrukciót hajtja végre a processzor
 - Létezik számunkra egy *adat folyam* is: a memória rekeszeknek és regisztereknek az a sora, mely az egymás utáni instrukciókban operandusként szerepelnek.
 - Az adat folyam elemei *adat állapotteret* határoznak meg.
 - Egy-egy instrukció végrehajtása *állapot változást* hoz.

Az instrukció folyam végrehajtása

- A program (az instrukció folyam, a kód) futása állapot átmenetek láncolatát hozza.
- A vezérlés állapot átmenet láncolat kulcsjellemezője a programszámláló regiszter egymás utáni értékei: a vezérlés menete (flow of control).¹
- A Neumann elvű gépekre jellemző ez az *egy vezérlés menet* (Single Instruction Stream) *egy adat folyamon* (on Single Data Stream): SISD²

A Neumann gép és az imperatív programozás

- Az imperatív nyelvekkel a vezérlés menetét manipuláljuk
 - Tedd ezt ezzel, utána ezt stb.
 - FORTRAN, C, Pascal, Basic
- Ezért az imperatív programozási paradigma jól megfelel a Neumann gépnek

Hiba és eseménykezelés a Neumann gépen

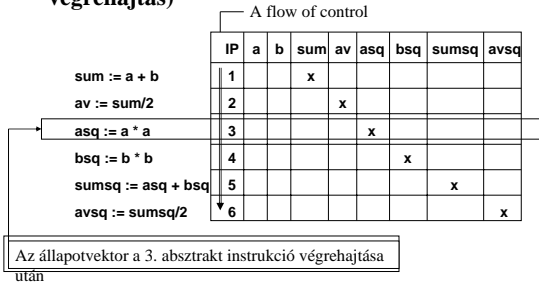
- Az eseményekhez kezelő (handler) instrukciófolyam tartozik
- Az esemény bekövetkeztekor a vezérlés menete ugorjon a kezelőre (a CPU állapot, a kontextus „lementése” után)
- A kezelés után (ha lehetséges) a vezérlés menete térjen vissza a „normál” instrukció folyamra, folytatódjon a processz futása (az állapot, a kontextus vissza-emelése után persze).
- Összegezve: a hiba és eseménykezelés a vezérlés menetének manipulálásával történik.

Egy más elvű gép: adatfolyam gép

- A Dataflow Machine (kontrasztként) ideája:
- szeparált processzorok minden operációra (operáció lehet: aritmetikai, logikai, függvényhívás stb.)
 - Az operációk (processzorok) várnak, míg operandusuk értéke előáll, utána adják eredményüket.
 - A processzorok (operációk) függetlenek. A legkorábbi lehetséges pillanatban adják az eredményüket.
 - Az operációk végrehajtásának sorrendje az adatfolyamból adódik.

Példa: adott a és b, kiszámítandó átlaguk és négyzeteik átlaga

Egy virtuális Neumann gép (szekvenciális végrehajtás)



Gyakran használt fogalmak

- **Virtualitás, virtuális (látszólagos)**
 - Valami, ami valóságosan nem létezik, de mégis úgy használhatjuk, mintha létezne
 - Pl. virtuális meghajtó, emulált terminál, virtuális gép stb.
- **Transzparencia, transzparens (átlátszó)**
 - Valami, ami ott van, de nem látjuk, nem vesszük észre, nem kell törődni vele, mert átlátszó. (Pl. az előbbi virtuális diszk-meghajtót a hálózaton át egy fájlserver biztosítja, akkor a hálózat transzparens, nem kell vele törődni.)
 - Világos, tiszta, nem titkolt ...

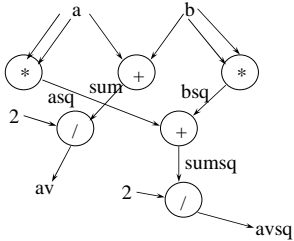
Példa: adott a és b, kiszámítandó átlaguk és négyzeteik átlaga
Neumann gép (szekvenciális végrehajtás)

IP	800	804	808	80C	810	814
600 a						
604 b						
608 sum	x					
60C av		x				
610 asq			x			
614 bsq				x		
618 sumsq					x	
61C avsq						x
800 ADD a,b,sum						
804 DIV sum, 2, av						
808 MUL a, a, asq						
80C MUL b, b, bsq						
810 ADD asq, bsq, sumsq						
814 DIV sumsq, 2, avsq						

Példa: adott a és b, kiszámítandó átlaguk és négyzeteik átlaga

Adatfolyam gép

sum := a + b
av := sum/2
asq := a * a
bsq := b * b
sumsq := asq + bsq
avsq := sumsq/2



A nyilak: neves, vagy név nélküli értékek.

A körök: az operációkhoz rendelt processzorok.

Hibakezelés:
explicit
hibaértékekkel

Példa: adott a és b, kiszámítandó átlaguk és négyzeteik átlaga

• Adatfolyam gép

- Van 6 processzor a 6 operációhoz,
- nincsenek "változók" (név, érték, típus, cím),
- névvel ellátott értékek vannak (a,b,asq,bsq,sum stb).
- A neves értékek nem definiálhatók át! Redefiníció esetén nem tudnák a processzorok, melyik értékre várjanak!
- A neves értékeknek típusa és *explicit hibaértéke* van! U.i. egy processzor mindenképp kell eredményezzen értéket, legfőképpen hibás eredményt! Kaphat hibás inputot is.

Számítógép - nyelvek - számítási modell

• A számítási modellek összetevői

- a számítás alapelemei
- a problémaleírás modellje
 - a leírás jellege és
 - módszere;
- a végrehajtás modellje
 - a végrehajtási szemantika
 - a végrehajtás kontrollja

A Neumann modell

- Az alapelemek: azonosítható entitásokhoz rendelt (típusos) adatok. (Változók, többszörös értékadás)
- Problémaeírás
 - procedurális/imperatív (lépésenként megadva ...)
- A végrehajtás modellje
 - a szemantika: állapotátmenet szemantika
 - a kontroll: közvetlen vezérlés (... a vezérlés menete ...)

Az adatfolyam modell

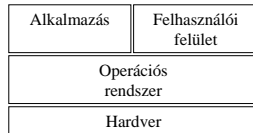
- Az alapelemek: azonosítható entitásokhoz rendelt (típusos) adatok. (Egyszeri értékadás)
- Problémaeírás
 - deklaratív (az operációk felsorolása pl. függvények használatával ...)
- A végrehajtás modellje
 - a szemantika: applikatív
 - a kontroll: adatfolyam vezérelt

Neumann elvű gép

- A gép fő részei, követelmények:
 - ALU, vezérlő egység, memória, perifériák,
 - 2-es számrendszer, elektronikus. Előtte?
- Tárolt program elv:
 - A tárban az adatok és a program is. Előtte?
 - Következményei: jók és rosszak.
- Automatikus működés (közvetlen vezérlés):
 - program szerint, állapotok, állapotátmenetek, a vezérlés menete, PC/IP szerepe.
- Babbage Analytical Engine: megfelel?

Számítógép: hardver és szoftver architektúra

- A legáltalánosabb SW architektúra
- Direkt futtatás, monitor, operációs rendszer
- Az OS fogalma
 - Kiterjesztett gép
 - Erőforrás menedzser



A rétegezettég

(layered architecture)

Egy réteg elrejtí az alatta fekvő rétegek részleteit.

Elegendő csak az alattad lévő réteg felületét (interface) ismerni

Operációs rendszer osztályozás

- Cél szerint: általános, cél
- HW "nagyság" szerint: PC, kis, nagy, szuper
- Processzorok, processzek, felhasználók száma szerint
- Időosztás szerint: szekvenciális, time sharing: kooperatív, beavatkozó
- Memóriamenedzselés szerint: valós, virtuális
- Fájrendszer implementáció szerint

Összefoglalás

- Bevezetés
- Egy kis história ...
- A Neumann elvű gép és az adatfolyam gép.
- Számítási modellek
