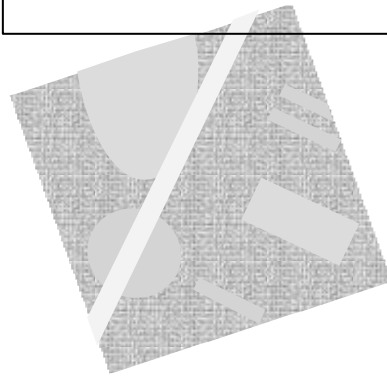


Luento 12

Yhteenveto



Keskeiset asiat
Mitä hyötyä tästä on?
Mitä seuraavaksi?
Kurssit?
Asiat?

23.8.2000

Teemu Kerola, K2000

1

Tavoitteet ⁽⁴⁾

- Ymmärtää tietokonejärjestelmän keskeiset piirteet sillä suoritettavan ohjelman näkökulmasta
- Miten tietokonejärjestelmä suorittaa sille annettua ohjelmaa?
- Minkälaista koodia suoritin ymmärtää?
- Mikä on käyttöjärjestelmän rooli?

23.8.2000

Teemu Kerola, K2000

2

Mitä hyötyä tästä on? ⁽²⁾

- Ohjelman suoritusnopeus perustuu suorittimen (CPU) suorittamiin konekäskyihin, ei pelkästään ohjelman korkean tason esitysmuotoon
- Ylemmän tason asioiden ymmärtäminen on helpompaa (mahdollista), kun ymmärtää alemman tason asiat

23.8.2000

Teemu Kerola, K2000

3

Keskeisiä asioita ⁽⁹⁾

- Järjestelmä kokonaisuudessaan, nopeuserot
- Esimerkkikone ja sen käyttö
- Konekielinen ohjelmointi
- Suoritin, rekisterit, väylät, muisti
 - konekäskyjen suoritussykli, keskeytykset
- Aktivointitietuepino, aliohjelmien toteutus
- Tiedon esitysmuodot (ohjelma vs. laitteisto)
- Välimuisti
- Prosessi ja sen toteutus (PCB)
- Ulkoinen muisti ja I/O laitteet
 - laiteajurit, laitekeskeytykset, levymuisti
- Ohjelmien suoritus järjestelmässä
 - käännös, linkitys, lataus, tulkinta, emulointi, simulointi

23.8.2000

Teemu Kerola, K2000

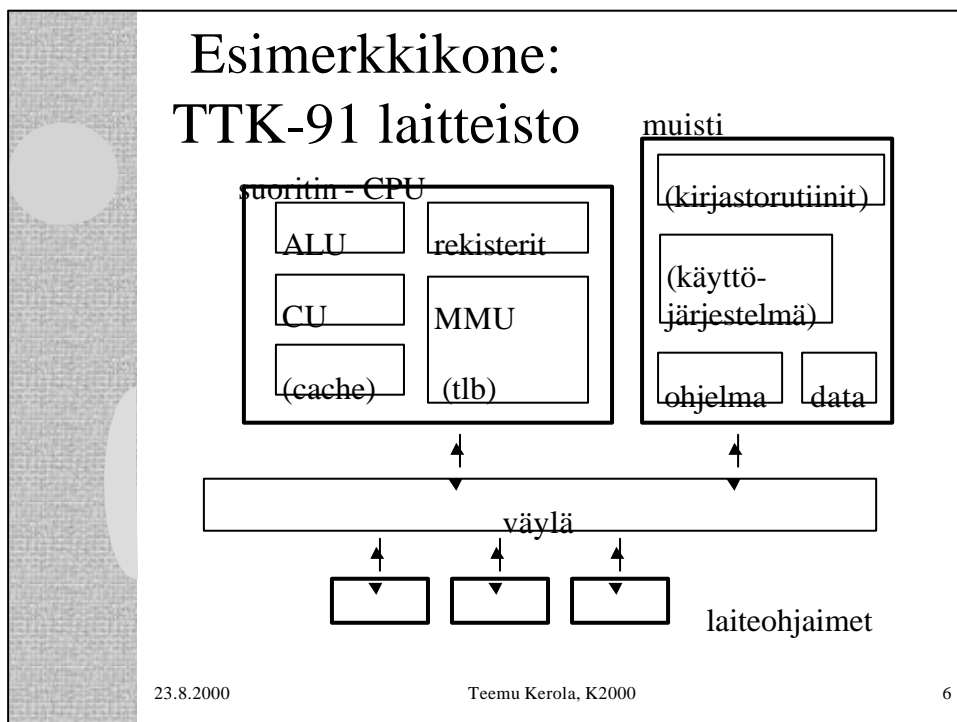
4

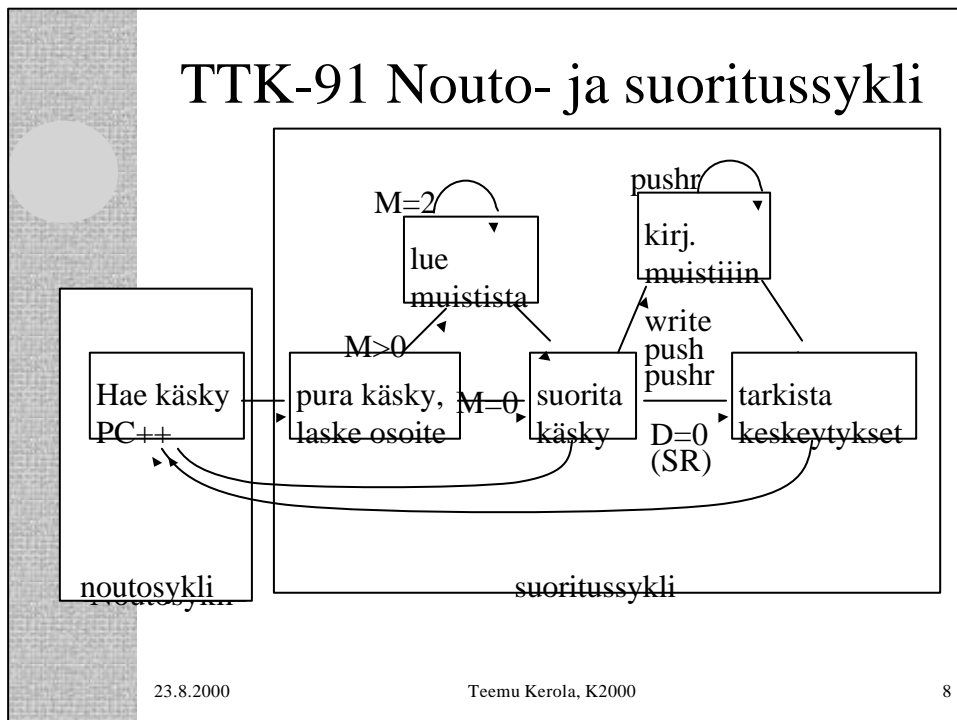
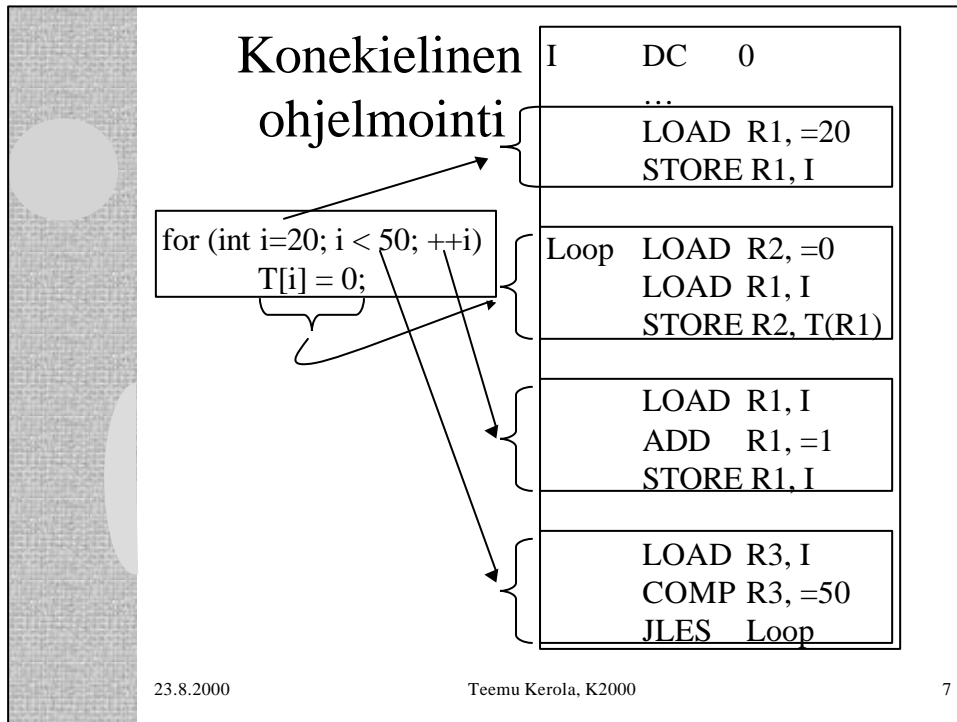
Nopeuserot: juustokakku

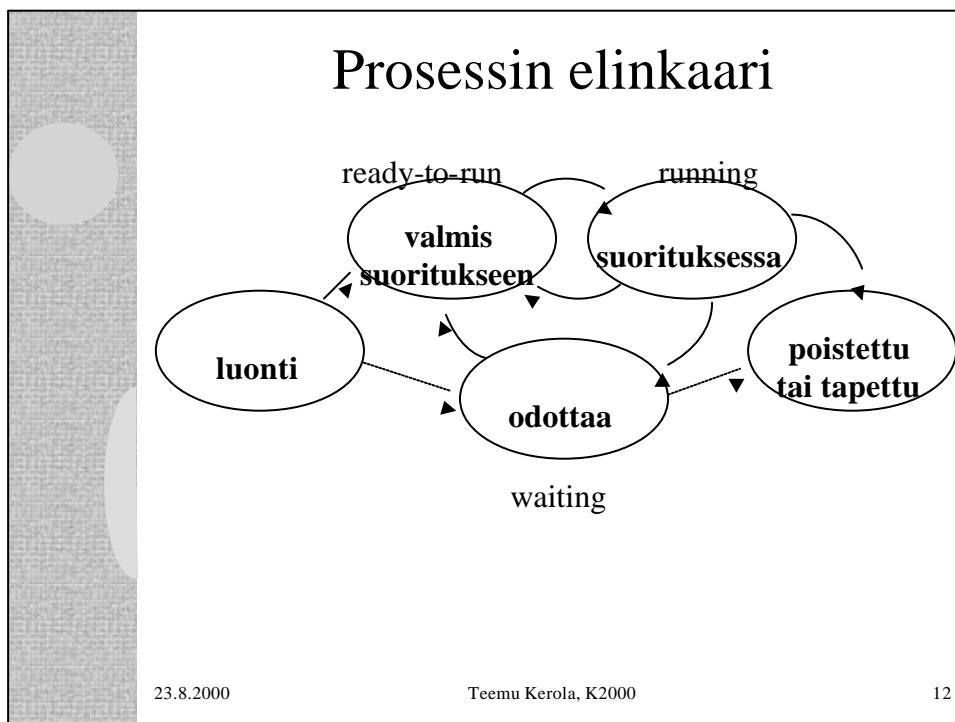
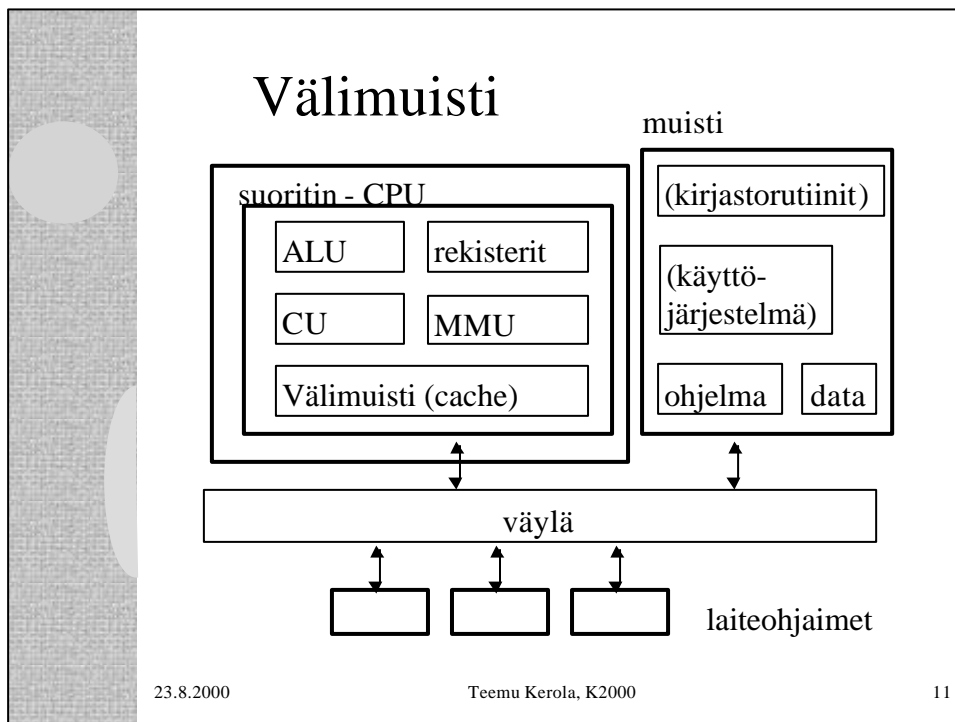
Rekisterien, välimuistin, muistin, levymuistin ja magneettinauhan nopeudet suhteutettuna juuston haku-aikaan juustokakkuja tehdessä?

Component / Storage	Access Time	Notes
käsi (hand)	0.5 sek	(rekisteri)
pöytä (desk)	1 sek	(välimuisti)
jääkaappi (refrigerator)	10 sek	(muisti)
kuu (moon)	12 pv	(levy)
Europa (Jupiter)	4 v	(nauha, ihminen)

23.8.2000 Teemu Kerola, K2000 5







Prosessit jonoissa ja PCB

R-to-R → 1056 → 1766 → Running → 0188

Disk1 → 0036 → 7654 → 9878

Timer → 0555

Msg from 1345 → 2222

Prosessin 9878
kuvaaja (PCB)

Vuoronanto:
valitse seuraava prosessi Ready-to-Run -jonosta ja
siirrä se suoritukseen CPU:lle
 (kopioi tämän prosessin suorittimen tila suorittimelle)

23.8.2000
Teemu Kerola, K2000
13

Laiteohjain ja laiteajuri

Muisti suoritin

väylä

data c/s
laiteohjain

laite laite

laiteohjainprosessi
(device controller)

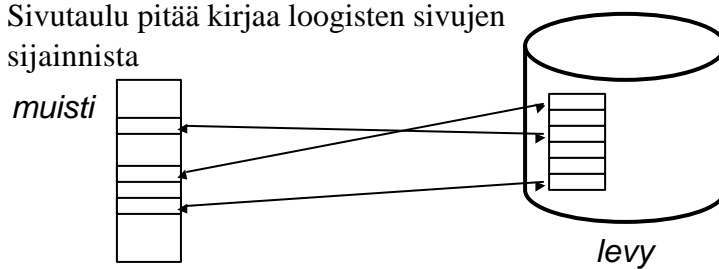
käyttäjäprosessi
 laiteajuri
 (KJ prosessi)

Ks. Fig 6.4 [Stal99]

23.8.2000
Teemu Kerola, K2000
14

Sivuttava virtuaalimuisti

- Looginen osoiteavaruus jaettu saman kokosiin sivuihin, esim. 1 KB
- Fyysinen muisti jaettu saman kokosiin sivuraameihin (page frame)
- Jokainen looginen sivu voidaan sijoittaa mihin tahansa (vapaaseen) sivuraamiin
- Sivutaulu pitää kirjaa loogisten sivujen sijainnista

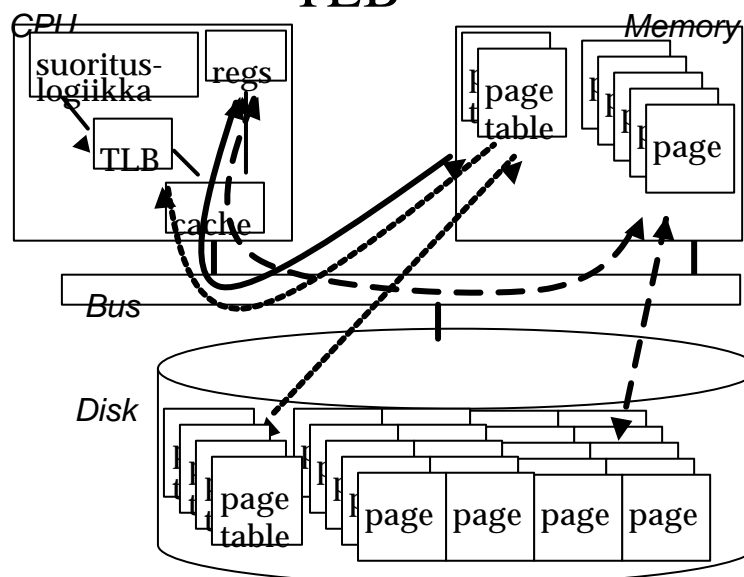


23.8.2000

Teemu Kerola, K2000

15

TLB



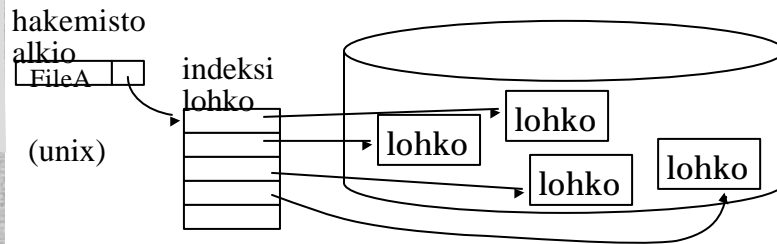
23.8.2000

Teemu Kerola, K2000

16

Levyjen käyttö

- Tiedosto koostuu useista lohkoista
 - lohko per sektori
- Levyn hakemistossa on tieto kunkin tiedoston käyttämistä lohkoista
 - luetaan lohkot annetussa järjestyksessä

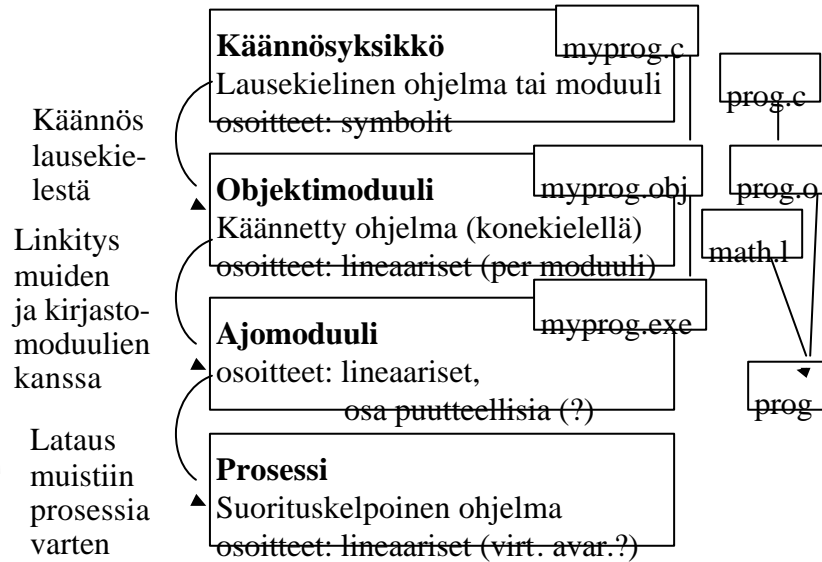


23.8.2000

Teemu Kerola, K2000

17

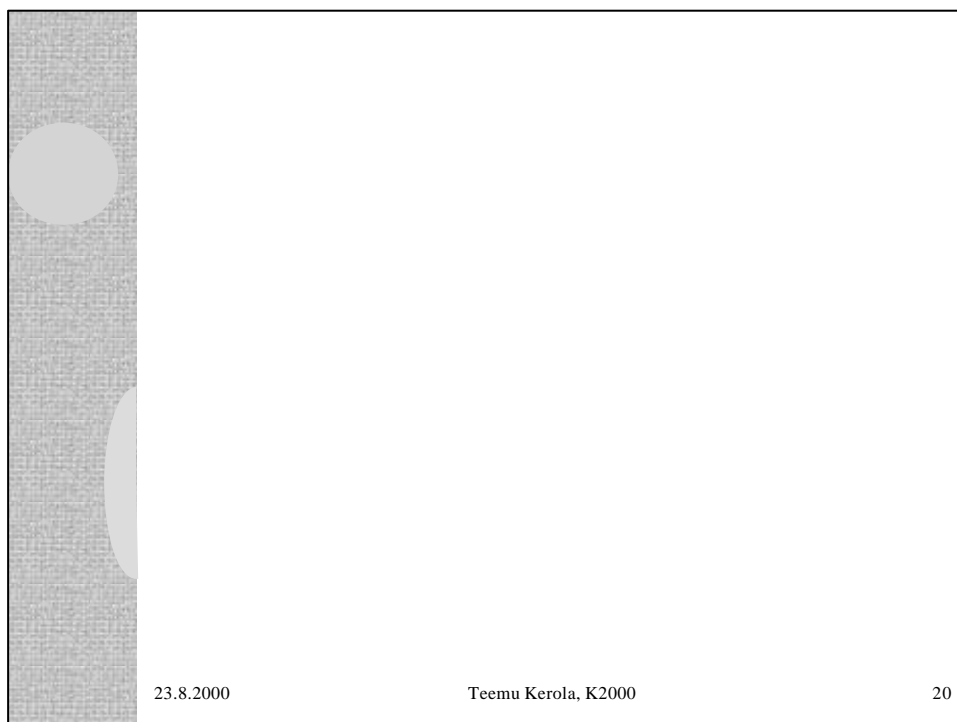
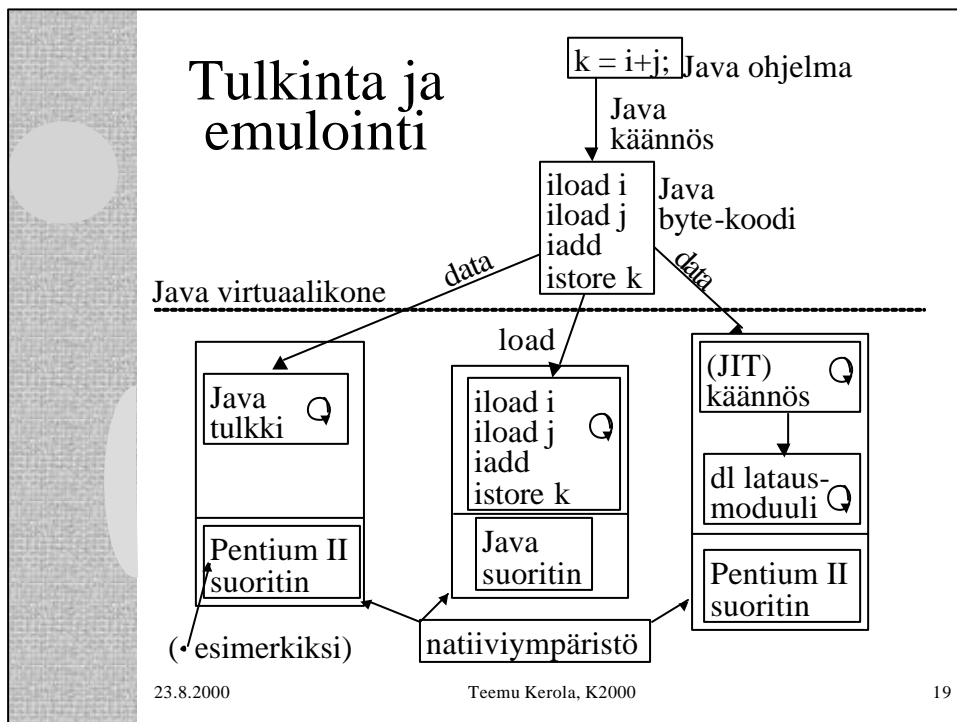
Lausekielestä suoritukseen

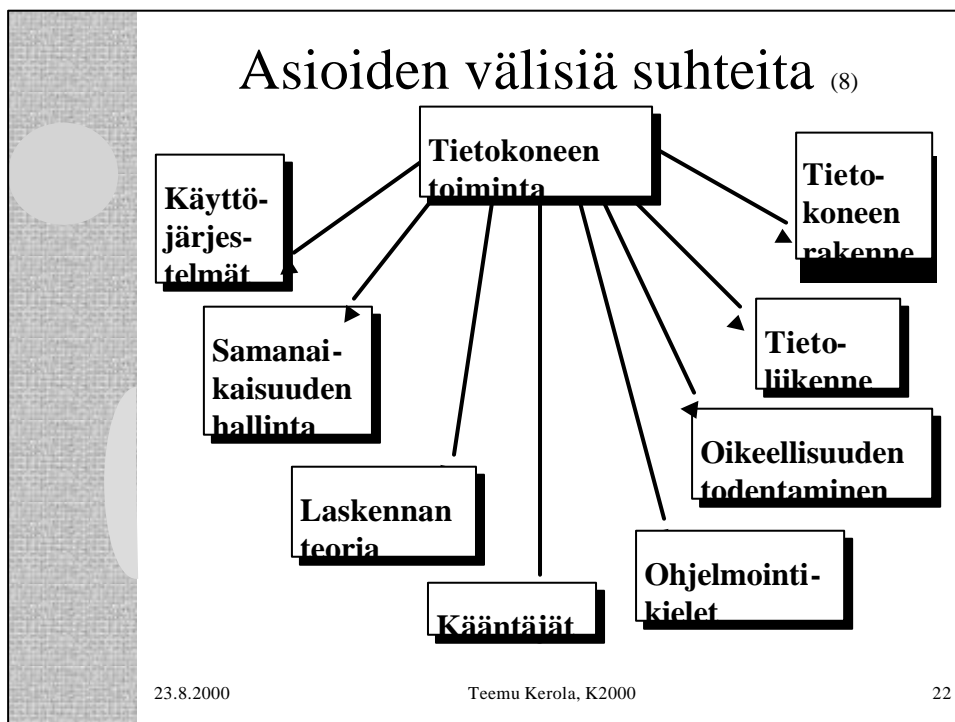
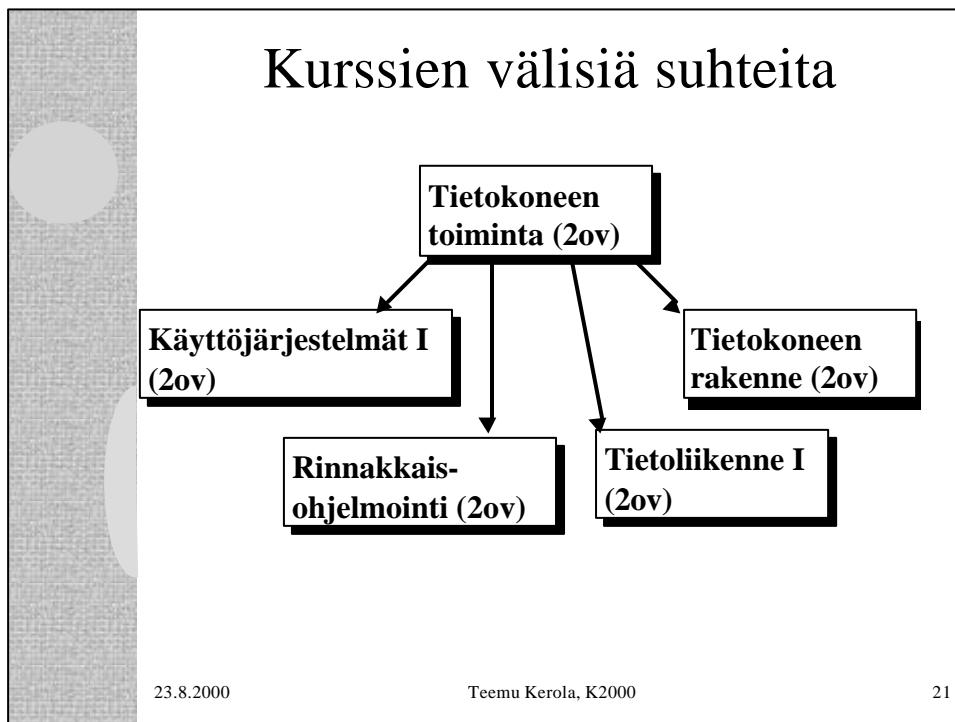


23.8.2000

Teemu Kerola, K2000

18





http://www.retroweb.com/apollo_retrospective.html

--
Luennon 12
ja
koko kurssin
loppu
--



<http://cookpages.com/MuriBeachcomber/>

23.8.2000 Teemu Kerola, K2000 23

Lisää tietoa kurssien sisällöstä

- kiinnostuneille
- erittäin kiinnostuneet voivat myös tutustua kurssien kotisivuihin ja kurssimappeihin talletettuun kurssimateriaaliin

23.8.2000 Teemu Kerola, K2000 24

Tietokoneen rakenne, 2 ov

- Yksi taso alaspäin TITOsta
- Sopiva 2. vuoden opiskelijalle
- Useissa korkeakouluissa yhdistetty TITOon
- ”Miten kellopulssi saa suorittimen suorittamaan konekäskyjä?”
- ”Miten suorittimen aritmetiikka on toteutettu?”
- Usea käsky on todellisuudessa suorituksessa samanaikaisesti
 - Miten tämä toteutetaan, mitä ongelmia siitä seuraa ja miten noita ongelmia ratkotaan?
- Jatkoa syventävällä tasolla
 - Tietokonearkkitehtuurit, 4 ov

23.8.2000

Teemu Kerola, K2000

25

TiKRra

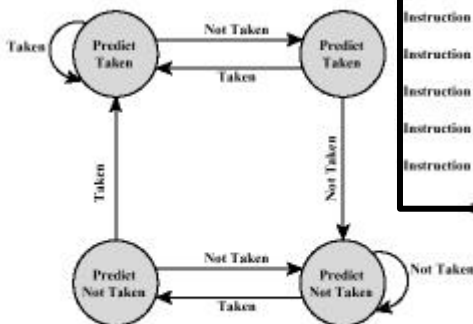
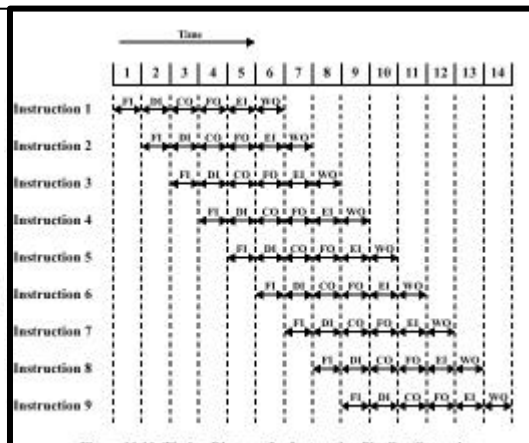


Figure 11.16 Branch Prediction State Diagram



[Sta199]

23.8.2000

Teemu Kerola, K2000

26

Käyttöjärjestelmät I, 2 ov

- Sopiva 2. vuoden opiskelijalle
- Käyttöjärjestelmän rooli yhden prosessin valvojana
- Täsmentää ja jatkaa TITOn käyttöjärjestelmien piirteiden esittelyä
- Samanaikaiset prosessit resurssien käyttäjinä
- Systemin resurssien jakelu
- Prosessien vuoronanto (skedulointi)
- Jatkoa perustasolla ja syventävällä tasolla
 - Käyttöjärjestelmät II, 2 ov
 - Käyttöjärjestelmämetodiikka, 3 ov

23.8.2000

Teemu Kerola, K2000

27

KJ ...

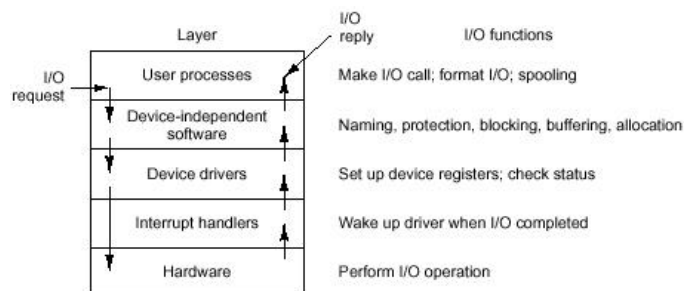


Figure 3-6. Layers of the I/O system and the main functions of each layer.

23.8.2000

Teemu Kerola, K2000

28

Tietoliikenne I, 2 ov

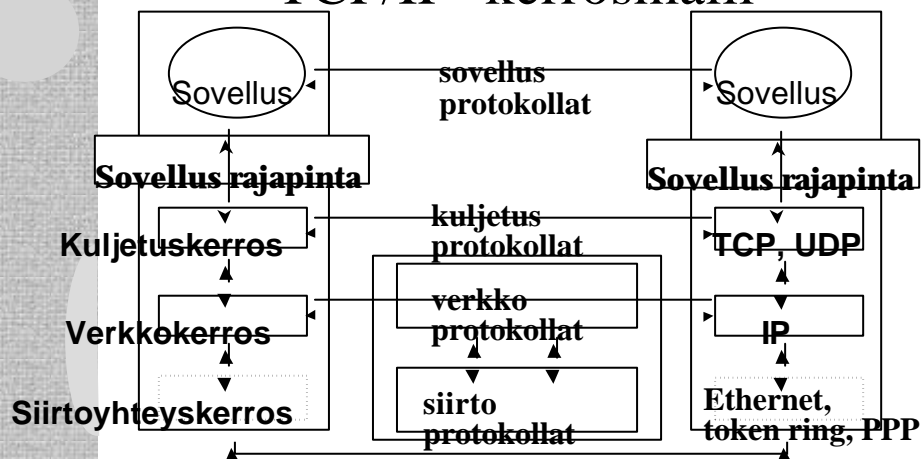
- Sopiva 2. vuoden opiskelijalle
- Tietokoneverkkojen peruspalvelut käyttäjälle ja sovelluksille
- Verkkojen tiedonsiirron perusvälineistö
- Verkkoarkkitehtuurin kerrosrakenne ja kunkin tason palvelut
- Jatkoa perustasolla ja syventävällä tasolla
 - Tietoliikenne II, 2 ov
 - Tietoliikennejärjestelmät, 3 ov

23.8.2000

Teemu Kerola, K2000

29

Tietoliikenne ... TCP/IP -kerrosmalli



23.8.2000

Teemu Kerola, K2000

30

Rinnakkaisohjelmointi, 2 ov

- Sopiva: 2. vuoden opiskelijoille
- Samanaikaisuuden aiheuttamat ongelmat
 - järjestelmä kaatuu ... miksi niin kävi?
- Samanaikaisuuden aiheuttamat vaatimukset systeemille
- Prosessien synkronointi eri tapauksissa
 - odottamalla vai prosessia vaihtamalla?
- Prosessien kommunikointi eri tavoin
 - yhteinen muistialue? viestit?
 - verkon ylitse?
- Jatkoa syventävällä tasolla
 - Hajautetut järjestelmät, 3 ov

23.8.2000

Teemu Kerola, K2000

31

RIO: synkronointiongelman ratkaisu Test-and-Set -käskyllä

- TAS Ri, L
(ttk-91:n
laajennus)

```

Ri := mem[L]
if Ri=1 then
  {Ri := 0, mem[L] := Ri, jump *+2}
  
```

- Kriittinen
vaihe

```

LOOP: TAS   R1, L   ) # L: 1 (vapaa) 0 (varattu)
      JUMP  LOOP   ^
      ...
      kriittinen vaihe
      ...
      LOAD  R1,=1
      STORE R1,L
  
```

- Toimiiko, jos tulee keskeytys pahalla kohtaa?

23.8.2000

Teemu Kerola, K2000

32

Ohjelmointikielten periaatteet, 4 ov

- Lähtötiedot: OKLPM, TiKi, ohjelmointilabrat
- Sopiva: 3. vuoden opiskelijat
- Ohjelmointikielten määrittelyn välineistö
- Erilaiset ohjelmointiparadigmat esimerkkikielten avulla
 - proseduraaliset kielet C, Pascal
 - oliokielet Smalltalk
 - funktionaaliset kielet Scheme, ML
 - logiikkaohjelmointikieliet Prolog
- Jatkoa syvemmillä tasolla:
 - ??

23.8.2000

Teemu Kerola, K2000

33

Ohjelmointikielten kääntäjät, 5 ov

- Lähtötiedot: OKLPM, ohjelmointilabrat
- Sopiva: 3. vuoden opiskelijat
- Ohjelmointikielten kääntäjien tyypit
 - rekursiivisesti etenevä jäsentelijä
- Kääntäjän osat
 - selaaja lex
 - jäsentelijä yacc
 - semantiikan analyysi
 - koodin generointi
- Jatkoa syvemmillä tasolla:
 - ??

23.8.2000

Teemu Kerola, K2000

34

Spesifioinnin ja verifioinnin perusteet, 2 ov

- Lähtötiedot: hajautuksen ja samanaikaisuuden problematiikka
- Sopiva: 3. vuoden opiskelijalle
- Mallinnetaan prosesseja siirtymäsystemeillä
- Automaattisen verifioinnin periaatteet
- Yksinkertaisia protokollien verifiointi
- Jatkoa syventävällä tasolla
 - Ohjelmien semantiikka, 3 ov
 - Automaattinen verifiointi, 3 ov

23.8.2000

Teemu Kerola, K2000

35

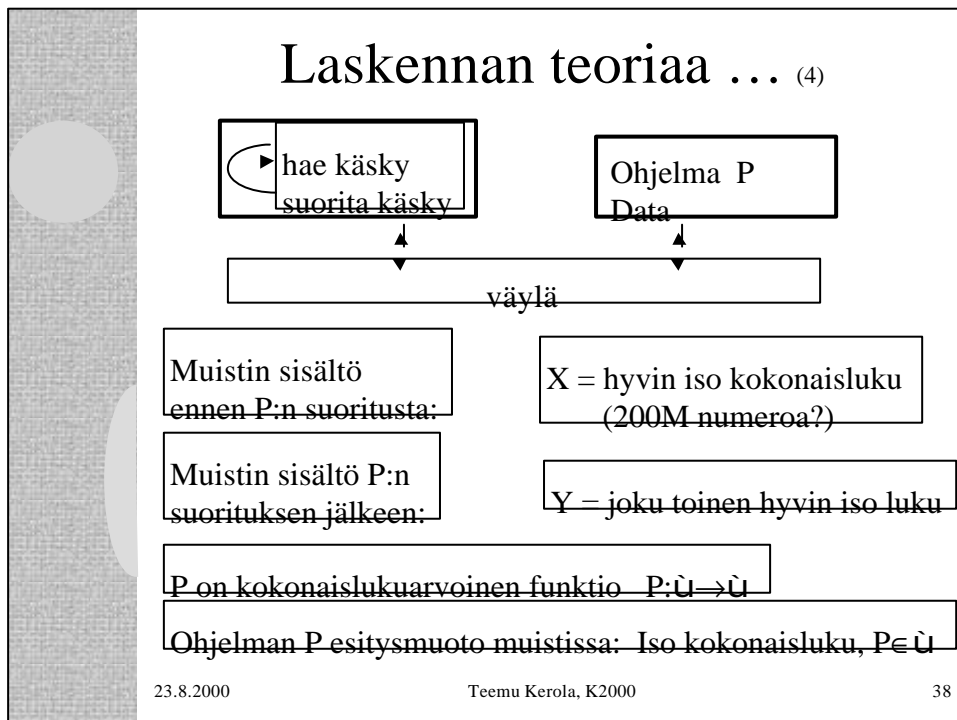
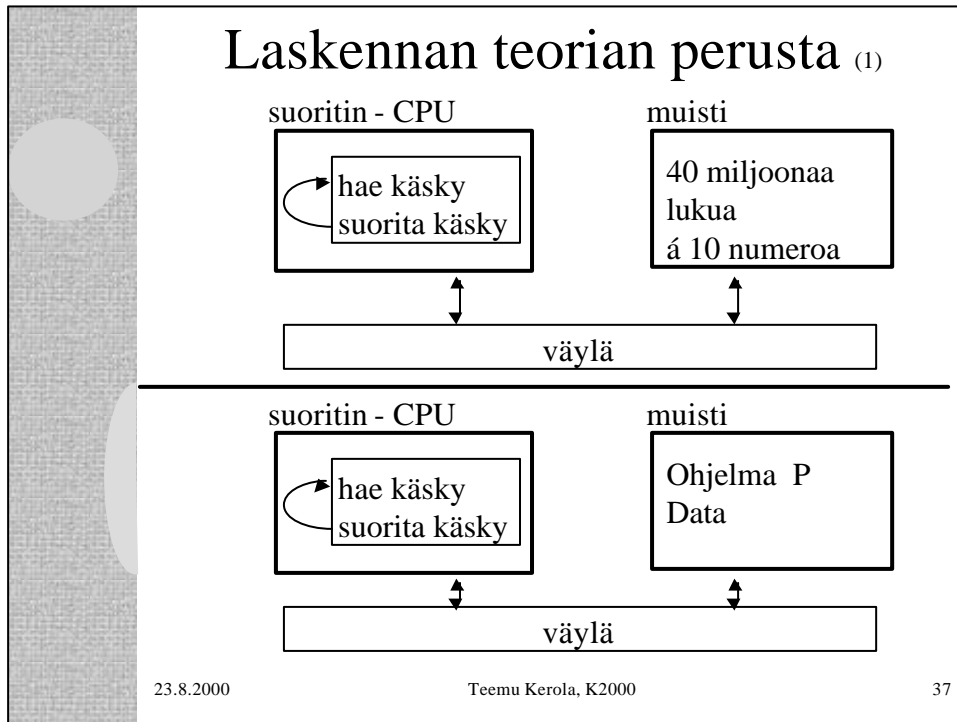
Ohjelmoinnin ja laskennan perusmallit, 2 ov

- Lähtötiedot: TiRa & matematiikkaa (Algebra I, ...)
- Sopiva: 1 vuoden opiskelijalle, joka on opiskellut jo matematiikkaa
- Laskennalliset ongelmat, niiden luokittelu
- Äärelliset automaatit ja säännölliset kielet
- Kielipit
- Turingin kone
- Jatkoa syventävällä tasolla
 - Laskennan teoria, 3 ov

23.8.2000

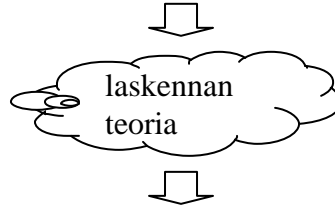
Teemu Kerola, K2000

36



Laskennan teoriaa ... (5)

- Mielivaltaisten ohjelmien ominaisuuksia voi päätellä kokonaislukujen ja niiden välisten funktioiden ominaisuuksista



- Todistettuja lauseita ohjelmien ominaisuuksista
 - pätevät kaikille tietokoneille
 - nyt ja tulevaisuudessa

23.8.2000

Teemu Kerola, K2000

39

Laskennan teoriasta ja algoritmianalyysistä todistettuja lauseita (3)

- Valitaanpa mikä tahansa aikaraja, niin aina on olemassa sellainen ongelma, että
 - (1) siihen on olemassa ratkaisu ja
 - (2) kaikki ongelman ratkaisevat ohjelmat vievät enemmän aikaa tai muistitilaa kuin ennalta annettu raja
- On olemassa ongelmia, että niitä ei voi ratkaista millään tietokoneella
- On olemassa suuri joukko tunnettuja vaikeita ongelmia, joista ei vielä tiedetä, kuinka vaikeita ne oikeastaan ovat

$$\boxed{P \stackrel{?}{=} NP}$$

23.8.2000

Teemu Kerola, K2000

40