

Informatika – Građa računala i princip rada

Prof. dr. sc. Tomislav Pribanić

Prof. dr. sc. Marija Seder

Izv. prof. dr. sc. Jurica Babić

Sveučilište u Zagrebu Fakultet elektrotehnike i računarstva

Vojni studijski program: Vojno inženjerstvo

Vojno vođenje i upravljanje

Računalo

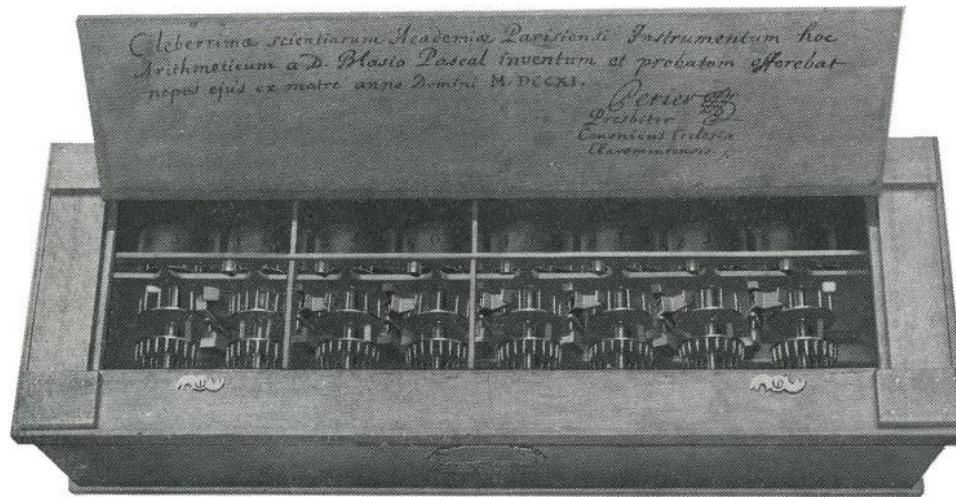
- uređaj za automatsko izvršavanje operacija (problema), a koje se mogu izraziti u numeričkom i/ili logičkom obliku
- rješavanje zadanog problema pretvaranjem svih relevantnih informacija u matematičke relacije korištenjem binarnog sustava (nule i jedinice)
- Informatika - izučava postupke i uređaje za automatski prijenos i obradbu podataka
- podatak \Leftrightarrow činjenica
- informacija \Leftrightarrow značenje pridruženo podatku

Podjela računala

- velika računala, radne stanice, osobna računala
- stolna računala, prijenosna računala, dlanovnici
- podjela prema tehnologiji izrade
- većina današnjih računala zasnovana je na von Neumannovoj arhitekturi

Nulta generacija računala

- mehanički strojevi za računanje, 1642 – 1945
- konfiguracija sustava zupčanika i njihovih odnosa
- B. Pascal, 1642: zbrajanje i oduzimanje

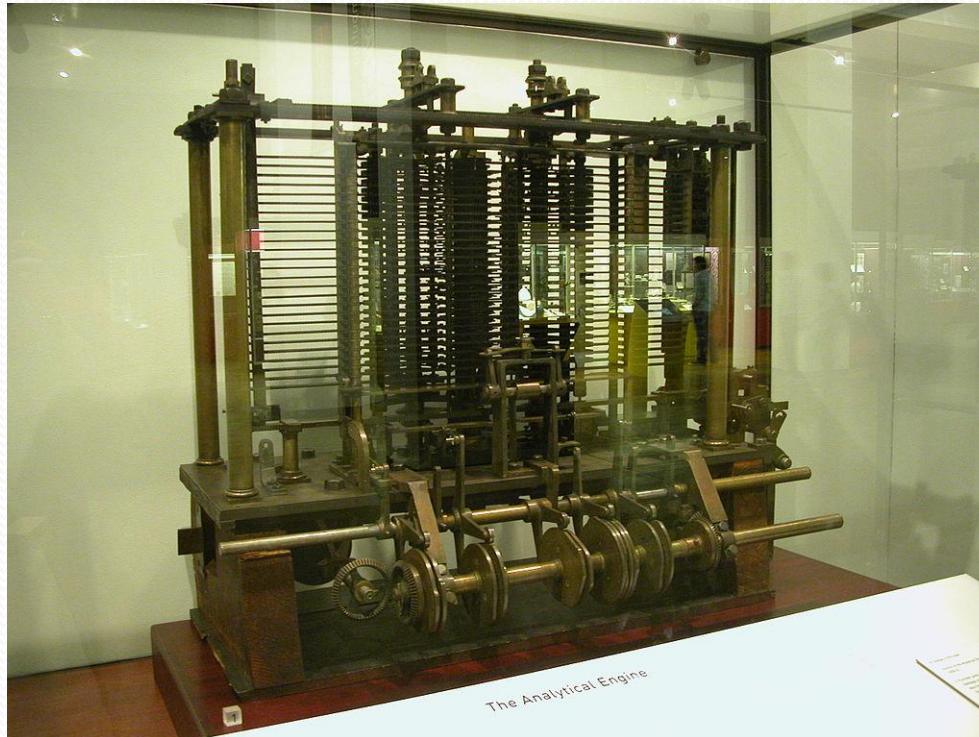


Analitički stroj

- 1673. G. W. Leibniz uz zbrajanje i oduzimanje, operacije množenja i dijeljenja (“džepni kalkulator” sa četiri funkcije)
- Leibniz: "It is unworthy of excellent men to lose hours like slaves in the labour of calculation which could safely be relegated to anyone else if machines were used."
- Charles Babbage, “otac računala”; 1822. izgradio *diferencijski stroj*
- 1834. započeo rad na novom konceptu: zamislio analitički stroj opće namjene: memorija, jedinica za računanje, *ulazna jedinica (čitač bušenih kartica) i izlazne jedinice (pisač i bušač kartica)* => programibilnost omogućena
- *tehnološki problemi, nikad dovršen* ☺

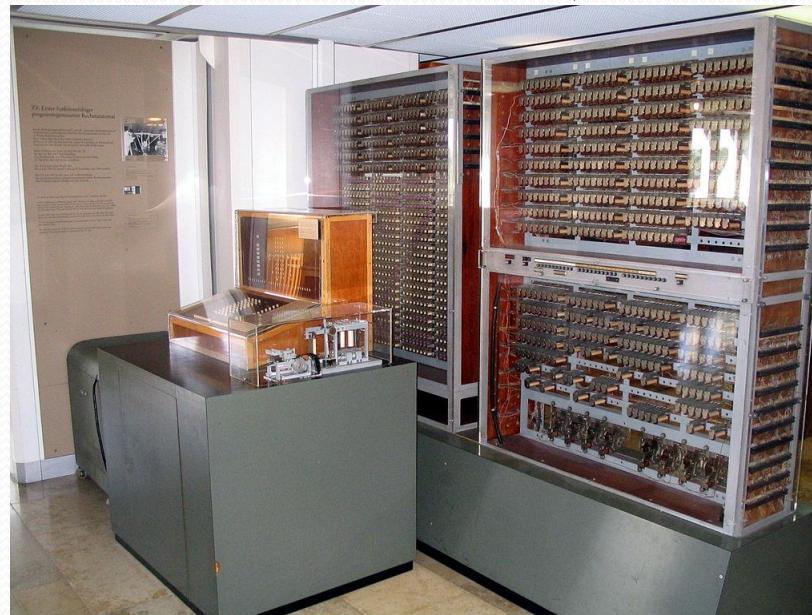
Rekonstrukcija Analitičkog stroja (Babbage)

- prvo programabilno računalo, opće namjene



Prvo elektromehaničko računalo

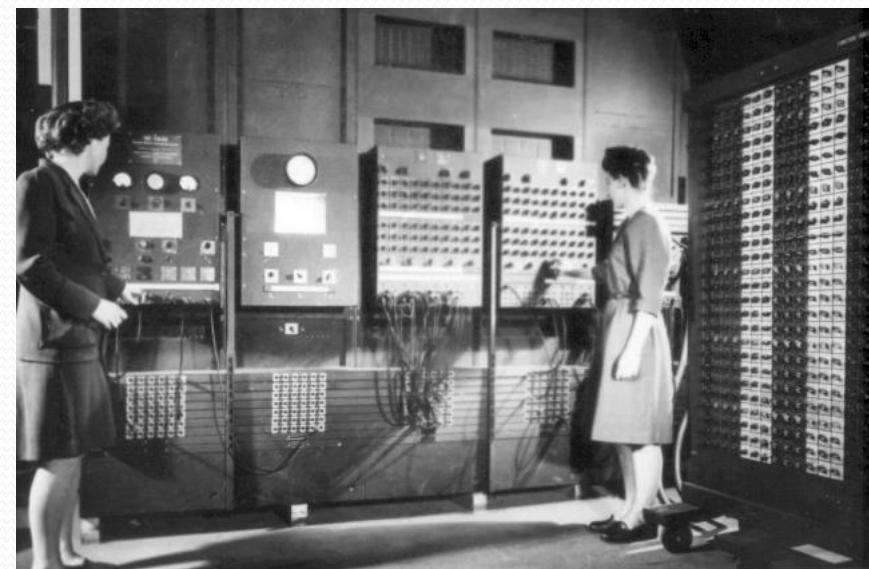
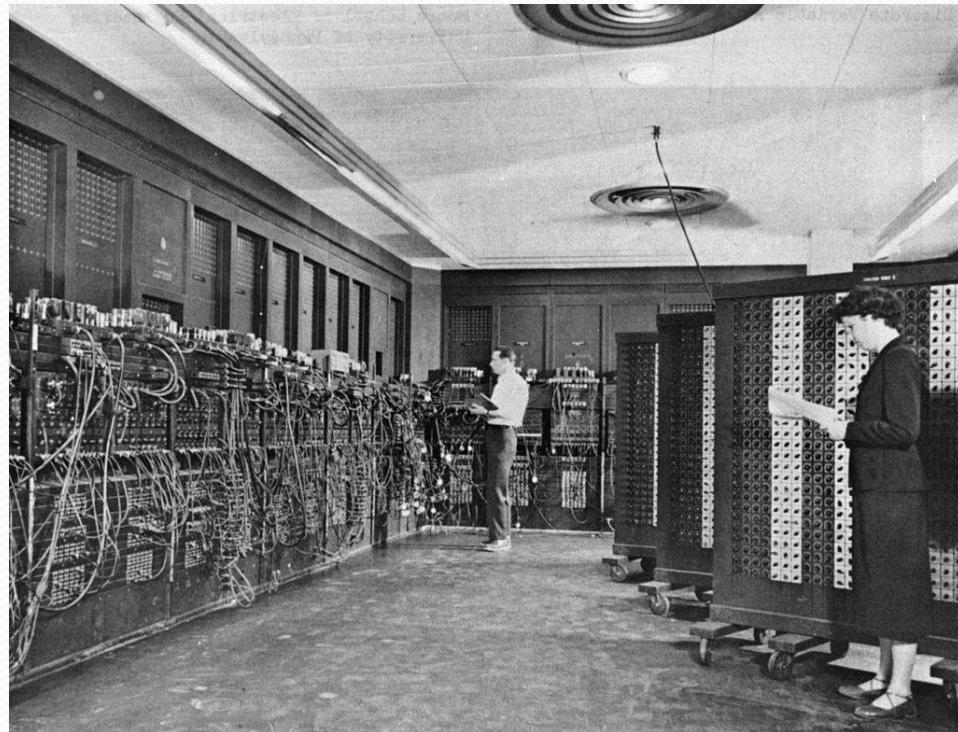
- prekretnica u tehnološkom smislu sa potpuno mehaničkih strojeva na elektromehaničke računske strojeve; stroj nazvan Z₁, K. Zusea (1934. – 1936.) koji se temeljio na elektromehaničkim relejima
- Zuse Z₃:
 - Berlin 1941
 - 2000 releja
 - takt 5-10Hz



Prva generacija računala

- računala s elektronskim cijevima, 1945 – 1955
- 1943. Engleska, razbijanje šifriranih poruka, COLOSSUS koje se smatra prvo elektroničko računalom
 - dvije tisuće elektronskih cijevi, sudjelovao i A. Turing.
- 1946 J. Mauchley i J. P. Eckert ENIAC (Electronic Numerical Integrator And Computer) računalo
- 18000 elektronskih cijevi i 1500 releja, oko 30 tona,
- žarenje elektronskih cijevi, anodne izvore i ventilatore za hlađenje bila mu je potrebna snaga od 140 kW!
- povijest suvremenih računala započinje s ENIAC-om

ENIAC



- upravljački panel

IAS računalo

- J. von Neumann sudjelovao u projektu ENIAC
- 1952. završio IAS (Institute for Advanced Study) računalo
- IAS svojim konceptima predstavlja osnovne temelje današnjih računala, tzv. *von Neumannov model računala*.
- godine 1953. tada mala tvrtka IBM započinje s proizvodnjom računala IBM 701

Druga generacija računala I

- tranzistor kao građevna komponenta, 1955 – 1965
- 1948. J. Bardeen, W. Brattain i W. Shockley izumili elektronički upravlјivu sklopku – tranzistor
- 1955. prvo računalo izgrađeno na temelju tranzistora TX-0 (*Transistorized eXperimental computer 0*) u *MIT Lincoln Laboratoryju*
- 1960. tvrtka DEC (*Digital Equipment Corporation*) na tržište plasira prvo malo računalo (miniračunalo) PDP-1 čija je cijena bila oko **120 tisuća dolara**

Druga generacija računala II

- 1965. DEC proizvodi 12-bitno miniračunalo PDP-8 čija je cijena bila samo 16 tisuća dolara. Tvrtka DEC prodala je preko 50 tisuća računala PDP-8
- bušene kartice i dalje prisutne kao sučelje za izmjenu podataka
- prvi prijelazi sa binarnog strojnog jezika na simbolički assembler
- rane verzije viših programskej jezika COBOL-a i FORTRAN-a

Treća generacija računala I

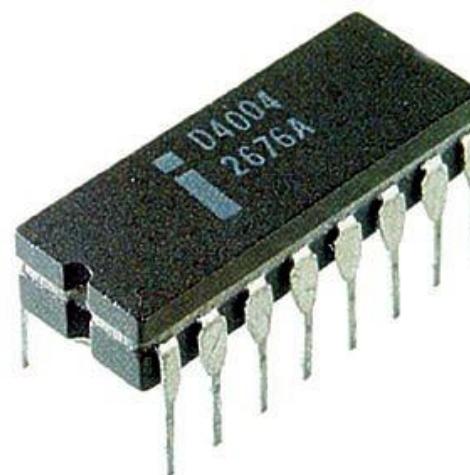
- računala sa integriranim sklopovima, 1965 – 1980
- integracija velikog broja tranzistora na komadiću silicija te oblikovanje u tzv. integrirani sklop ili čip
- manja, brža i jeftinija računala, višeprogramska rad
- UNIVAC 1100, PDP-11
- 1971. pojavljuje se na tržištu prvi 4-bitni mikroprocesor Intel 4004 (predviđen kao kalkulatorski čip)

Treća generacija računala II

- 1972. prvi 8-bitni mikroprocesor opće namjene Intel 8008
- 1974. pojavljuje se druga generacija 8-bitnih mikroprocesora čiji su tipični predstavnici Motorola 6800 i Intel 8080
- započinje revolucija na području računala
- računala postaju šire dostupna
- uporaba tastatura i monitora kao sučelja za razmjenu podataka

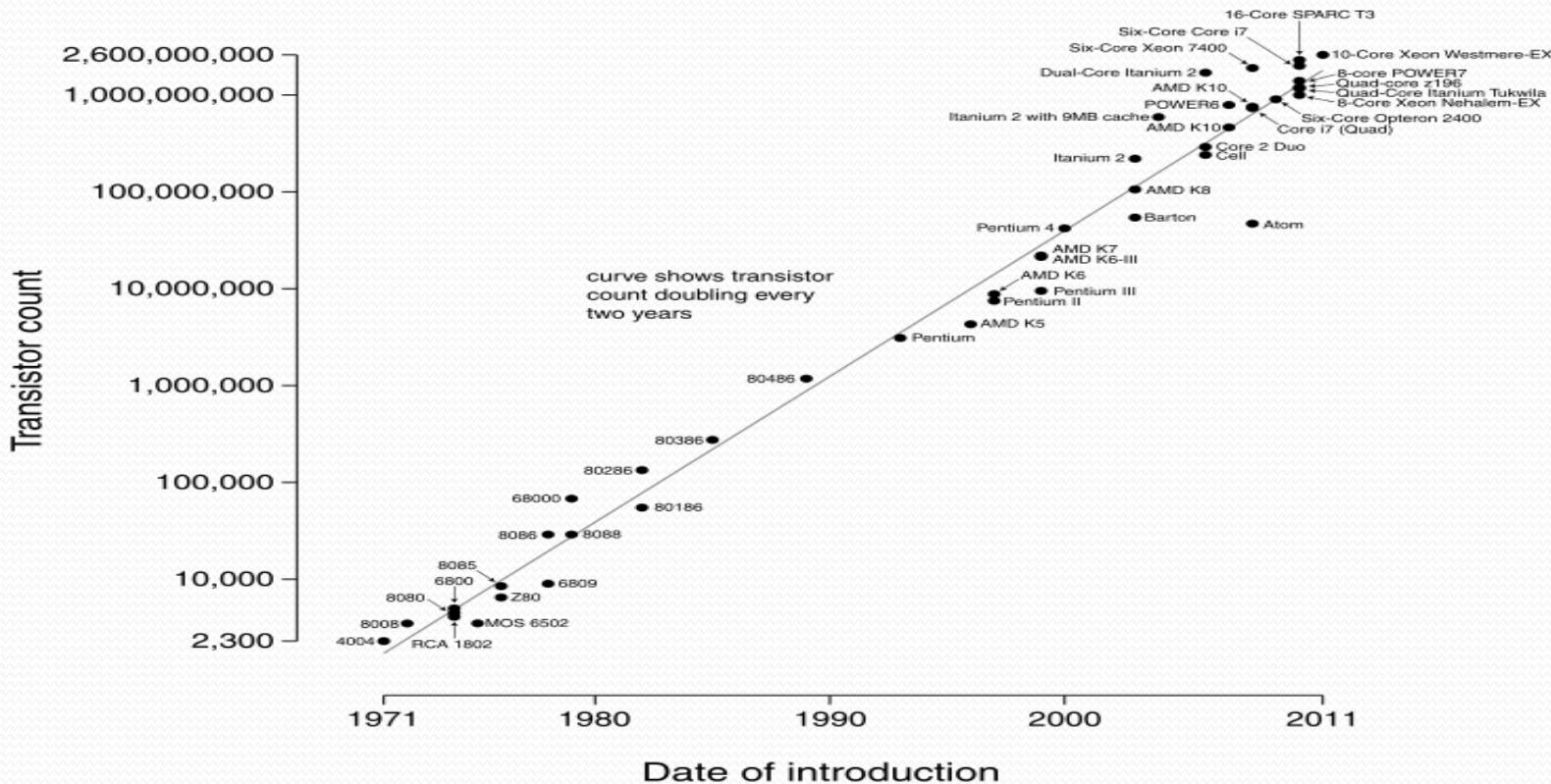
Četvrta generacija računala I

- sklopovi vrlo visokog stupnja integracije VLSI (Very Large Scale Integration), 1980 - ?
- realizacija integriranih sklopova sa stotine milijuna tranzistora
- Mooreov zakon: broj tranzistora na čipu se udvostručuje svakih 18 – 24 mjeseca
- osnova generacije, 1971. prvi Intel-ov mikroprocesor 4004



Moorov zakon

Microprocessor Transistor Counts 1971-2011 & Moore's Law



- 1965. predviđanje Gordon E. Moore, suosnivača Intela

Četvrta generacija računala II

- pojava osobnih računala
- 1981 IBM-ova osobna računala (Intelovom mikroprocesoru Intel 8088 mikroprocesor)
- 1984 Apple-ov the Macintosh, Commodore, Amiga i Atari (ne Intelovi procesori)
- mikroprocesori - pored ugradnje u računala opće namjene, ubrzani razvoj tzv. ugrađenih računalnih sustava za videoigre, mobitele, perilice, laserske pisače, automobile
- pojava miša, grafičkog korisničkog sučelja (GUI)

Peta (šesta) generacija računala

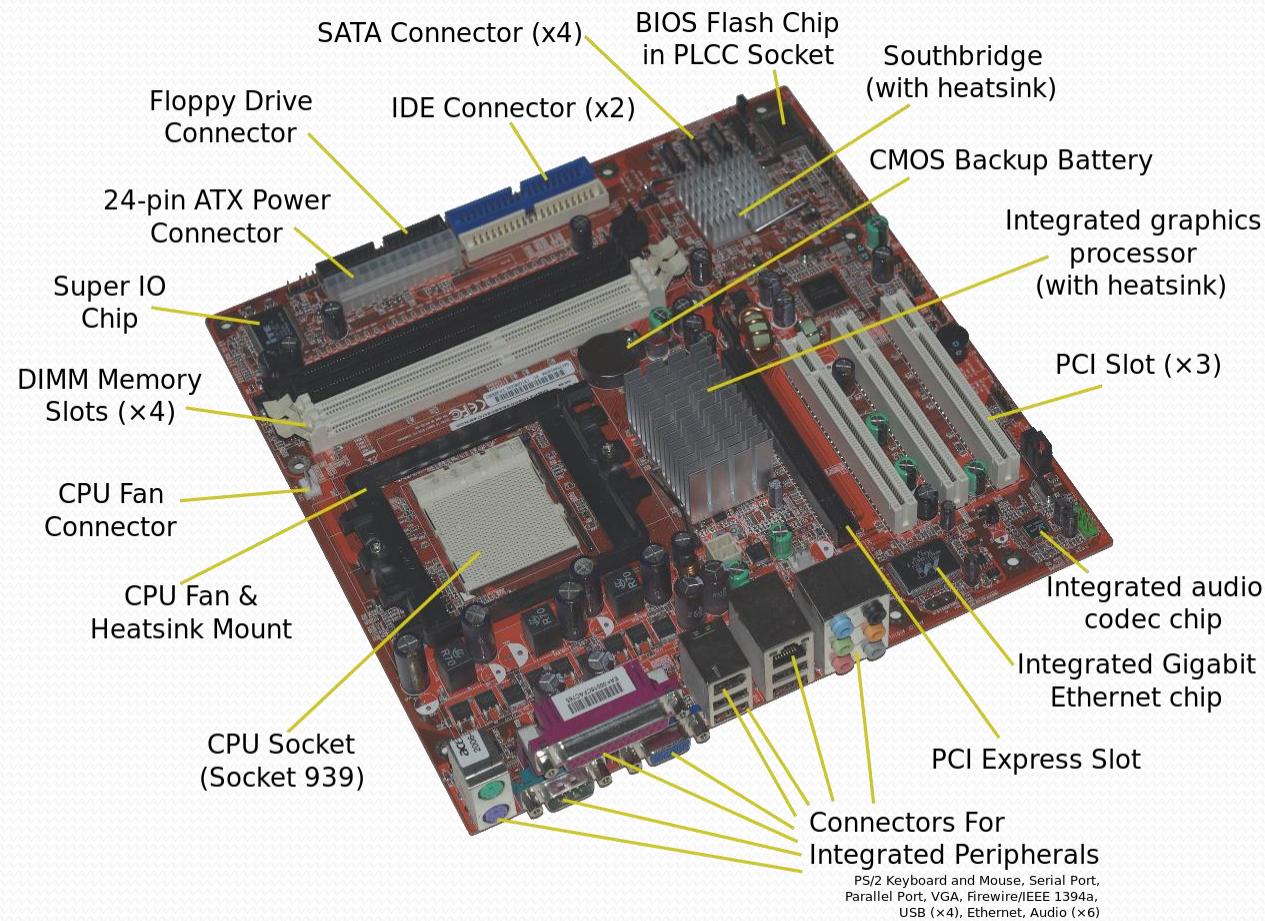
- koncept uporabe umjetne inteligencije 1980- danas
- neuronske mreže – reprodukcija živčanih veza
- prepoznavanje glasa/upravljanje glasom
- robotika
- sposobnost (samo)učenja
- nanotehnologije

Osnovni dijelovi računala

- matična ploča
 - grafička kartica
 - zvučna kartica
 - mrežna kartica
 - CPU središnja procesorska jedinica
 - ...
- ROM/RAM memorija
- tvrdi disk
- CD/DVD pogon
- sabirnice
- sučelja (USB, FireWire, Serijski/paralelni port...)
- napajanje

Matična ploča

- centralni dio sklopoljja u računalu na koju se priključuje ostalo sklopoljje



Grafička kartica

- AGP, PCI, PCI-Express sučelje
- Nvidia, ATI
- digitalno/analogno sučelje
- GPU vs. CPU
- memorija



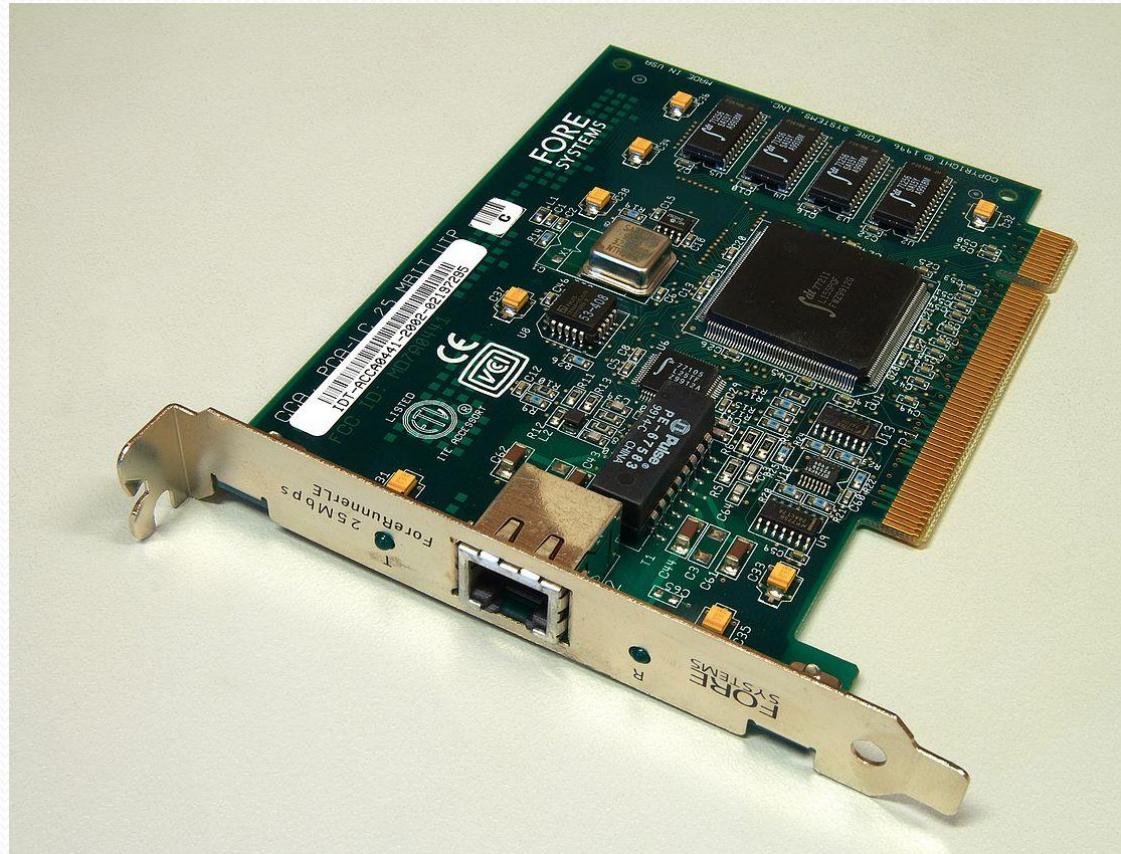
Zvučna kartica

- obrađuje ulazno izlazne audio signale



Mrežna kartica

- omogućava spajanje računala na računalnu mrežu
(Ethernet, Wi-Fi)



CPU centralna procesorska jedinica



- „mozak” računala
- mikroprocesor (čip)

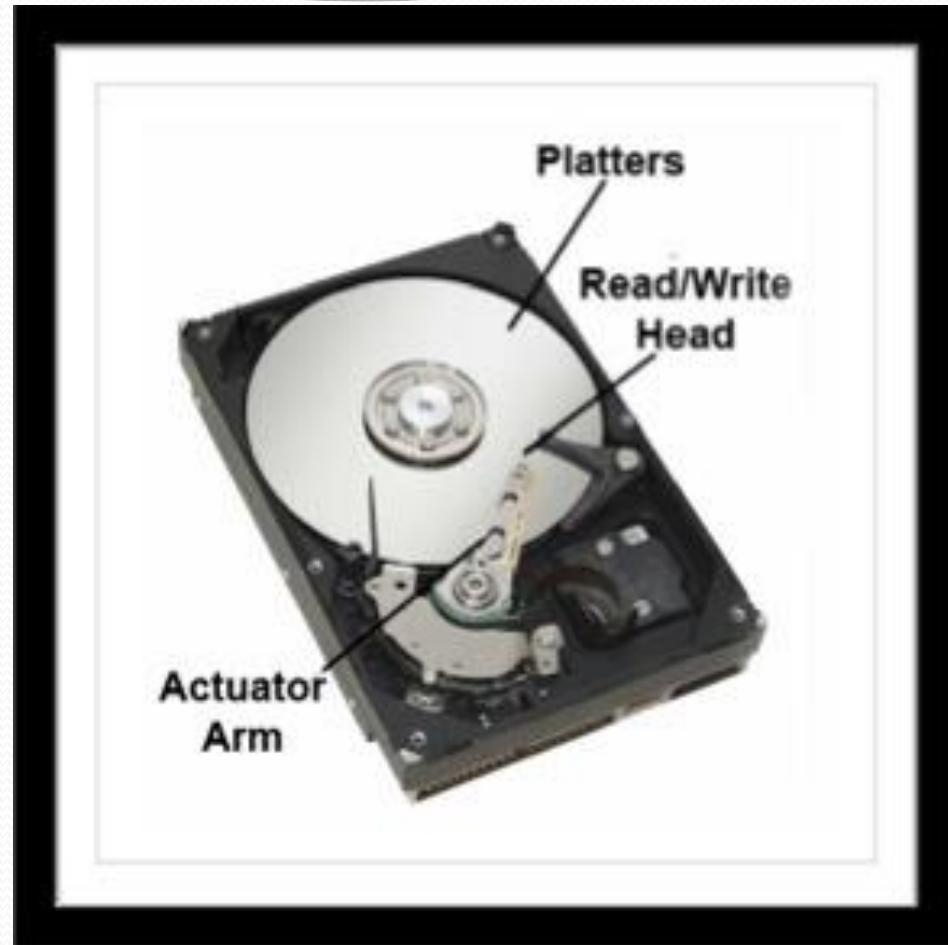
ROM/RAM

- ROM read only memory: zapis jednom upisan u memoriju zapis u memoriji se mijenja relativno rijetko (teško) ili nikada
- RAM (kratica od Random Access Memory - *memorija s nasumičnim pristupom*) je oblik tzv. primarne računalne čijem se sadržaju može izravno pristupiti
- upisuju se aktivni programi te informacije potrebne za trenutačan rad računala



Tvrdi disk

- sekundarna memorija
- metalne (staklene) kružne ploče, presvučene feromagnetskim slojem se vrte oko osi
- magnetske glave lebde iznad ploča te čitaju/upisuju podatke
- 1980. 1GB, 250kg, 40000 US dolara
- 2010. 3TB ⇔ 3x1024GB kapacitet



- SSD solid state disk
- „disk“ koji koristi poluvodičku tehnologiju (IC) za pohranu podataka; nema mehaničkih dijelova

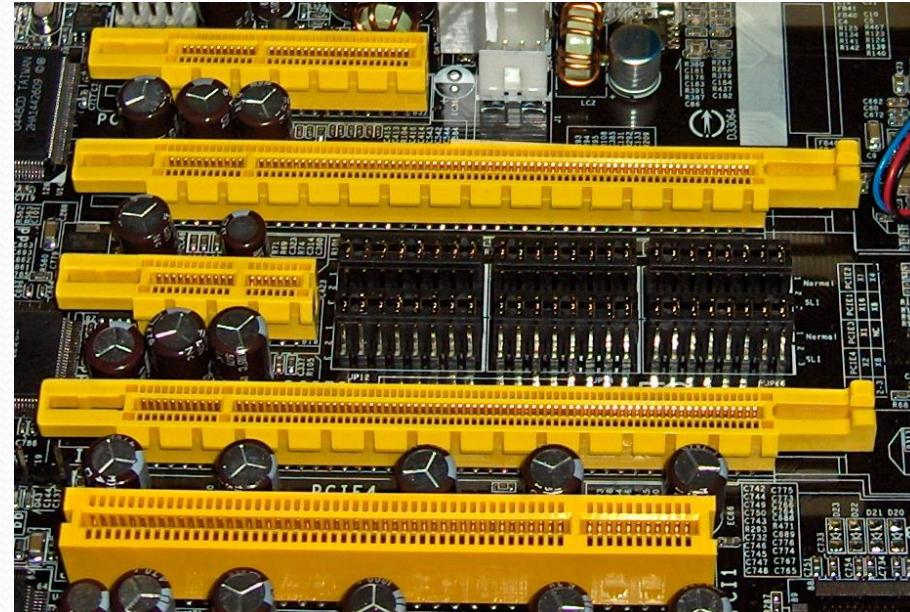
CD/DVD pogon

- laserski snop za upis/čitanje podataka (optički disk)



Sabirnica

- komunikacijski sustav (skupina vodiča) za razmjenu podataka između komponenti računala i/ili više različitih računala
- adresna, kontrolna i podatkovna sabirnica
- krajevi sabirnica - odgovarajući utori za povezivanje komponenti



VON NEUMANNOV MODEL RAČUNALA I

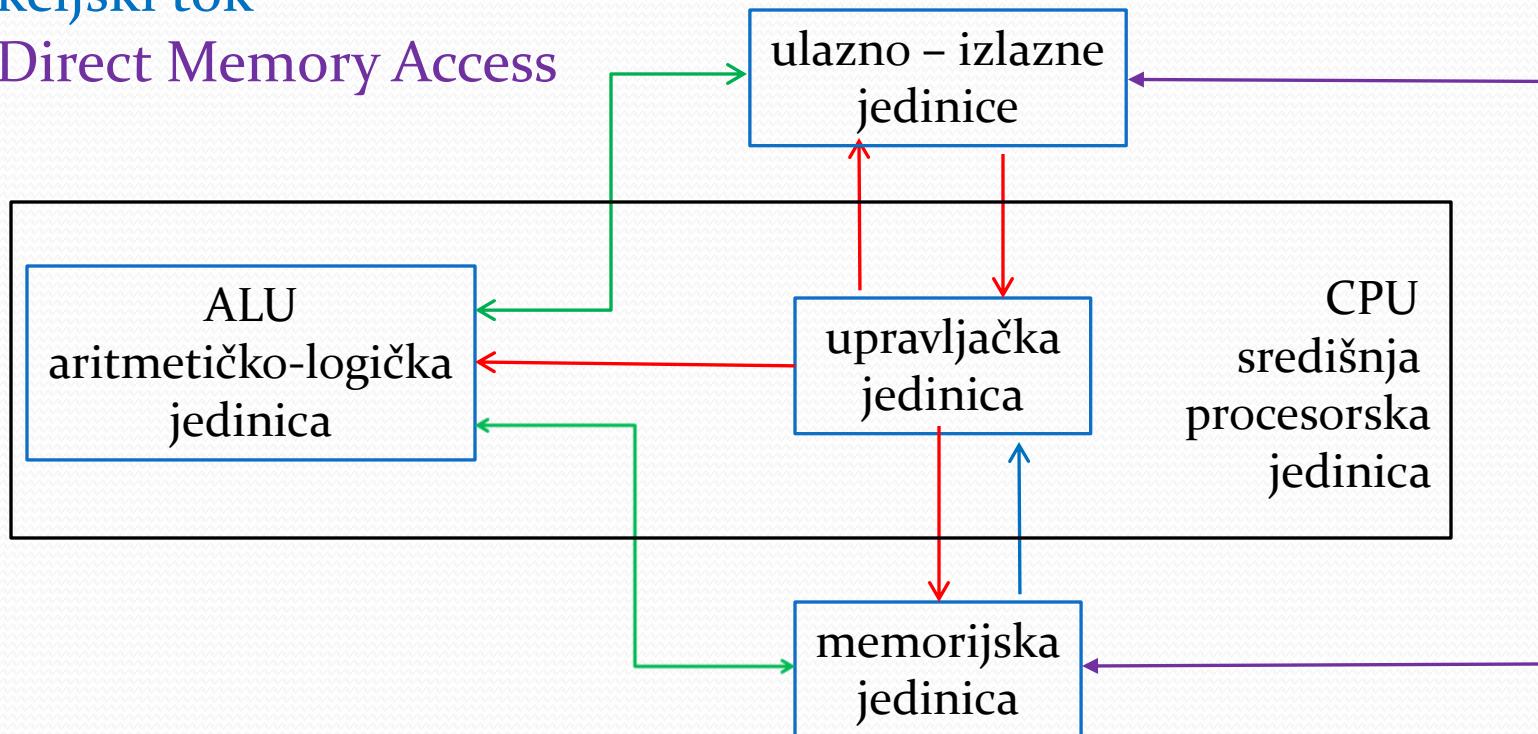
- 1946. *Preliminary Discussion of the Logical Design of an Electronic Computing Instrument*, A. W. Burksa, H. H. Goldsteina, von J. Neumanna
- osnova arhitekture nadolazećih računala:
 - računalo treba biti opće namjene
 - zahtjeva se automatizirani rad bez intervencije korisnika
 - posebni dijelovi za aritmetiku, memoriju, kontrolu te korisničko sučelje
 - mogućnost pohrane u memoriju ne samo podataka već i instrukcija (naredbi) programa

VON NEUMANNOV MODEL RAČUNALA II

- nametnula se potreba pohrane naredbi instrukcija također putem brojeva
- mora postajati način razlikovanja brojevnih zapisa u memoriji koji predstavlja podatke od onih koji predstavljaju naredbe programa te mogućnost tumačenja i izvršenja naredbi – upravljačka jedinica
- budući da uz aritmetičke operacije treba moći izvoditi i logičke operacije (I, ILI, NE) aritmetička jedinica postaje aritmetičko-logička jedinica

VON NEUMANNOV MODEL RAČUNALA III

- upravljački tok
- podatkovni tok
- instrukcijski tok
- DMA Direct Memory Access



VON NEUMANNOV MODEL RAČUNALA IV

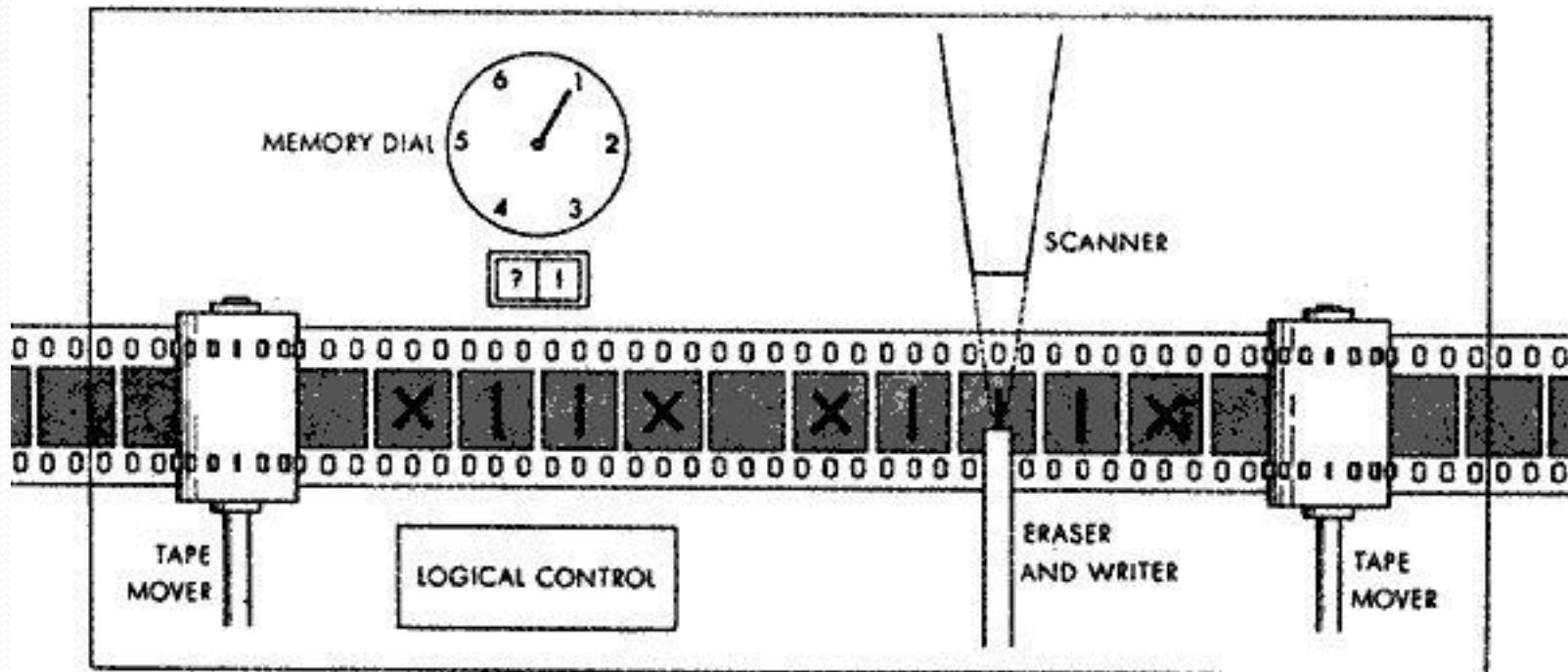
- aritmetičko logička jedinica: zbrajalo, akumulator, sklop za posmak, množenje/dijeljenje (?)...
- upravljačka jedinica: dekodira strojne instrukcije te generiranjem odgovarajućih upravljački signala pokreće sklopove u pojedinim funkcijskim jedinicama
 - strojna instrukcija – izvršenje programa: operacijski kod (operacija) i adresno polje (podatak)
- memorijska jedinica: adresiranje/upis/čitanje podataka
- ulazno-izlazne jedinice

Turingov stroj I

- 1936, Alan Turing
- apstraktni izvršitelj programa
- simulacija logike izvođenja programa (algoritma)
- analiza izračunljivosti algoritama
- komponente:
 - beskonačna traka (memorija) podijeljena u ćelije (segmente). Svaka ćelija sadrži jedan od konačnog broja simbola.
 - glava za čitanje/upis simbola u ćelije trake
 - registar stanja za zapis jednog od konačnog broja stanja
 - tablica instrukcija (prijelaza stanja) - određuje brisanje/upis novoga simbola, (ne)pomicanje glave za čitanje lijevo/desno, prijelaz u novo stanje

Turingov stroj II

- <http://blog.computationalcomplexity.org/2003/01/foundations-of-complexity-lesson-12.html>

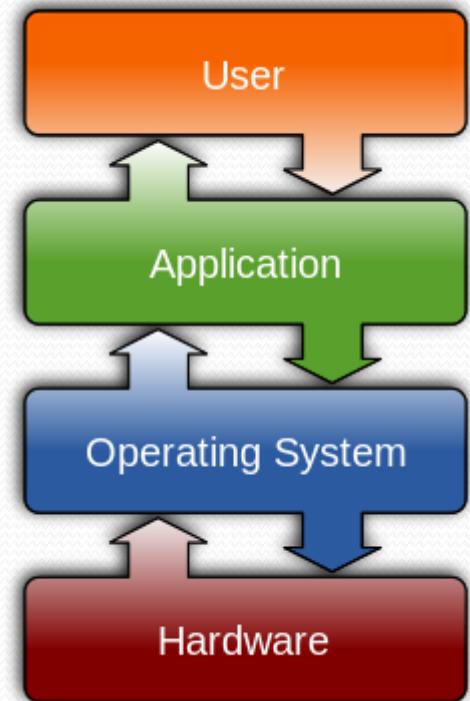


???ware

- hardware – fizički dio računalnog sustava
- software – programi (podaci) računalnog sustava
 - korisnički softver
 - sistemski softver
- firmware – softver (podatci) ugrađen u hardverski uređaj
- malware – programi za ometanje normalnog rada računalnog sustava
- humanware – softver & hardver dizajniran za interakciju sa korisnikom (www.humanware.com/)
- freeware, shareware, ...ware

Operacijski sustavi (OS)

- skup osnovnih programa koji upravljujaju sklopoljem računala radi ostvarivanja osnovnih funkcija računala: ulaz, memoriranje, obrada i izlaz podataka itd.
- Android, BSD (U. California, Berkeley), iOS, Linux, OS X
- Microsoft Windows, Windows Phone
- komercijalni vs. besplatni



Motivacija za OS

- nulta generacija računala - program je imao potpunu kontrolu na strojem, nema potrebe za OS-om
- sve niže operacije (npr. ispis) sastavni dio programa
- programer - nužno poznavanje sklopolja računala, česte sklopovske prepravke na samome računalu
- prvi OS se razvijaju u prvoj, a pojavljuju u drugoj generaciji (> 1955. godine) računala
- pojava biblioteka zajedničkih funkcija

Tipovi operacijskih sustava

- real time OS – aplikacije koje zahtijevaju odziv i izvršenje u realnom vremenu
- multiuser OS- pristup računalnom sustavu od strane više korisnika
- multitasking OS – istovremeno izvršavanje više aplikacija
- distributed OS – usklađuju rad grupe računala kao da se radi o jednom računalu
- embeded OS – za rad sa ugradbenim računalnim sustavima

Komponente OS-a I

- kernel – sprega između korisničkih aplikacija i hardverskih dijelova
 - pokretanje aplikacije => dodjeljivanje procesa (memorija, prioritet)
 - prekidi (interrupts) – mehanizam odaziva OS na različite zahtjeve programa i/ili hardvera
 - mod rada (pristupa resursima) – suprevisor (neograničen) vs. protected
 - dodjeljivanje memorije – virtualna memorija, swaping
 - višezadaćnost (multitasking) – obavljanje više zadaća (programa) istovremeno, time-sharing, preemptive vs. cooperative multitasking

Komponente OS-a II

- pogonski programi (device drivers) – transparentna interakcija sa hardverom „jezikom“ doličnog hardvera
- mrežna komunikacija – podržavanje raznih mrežnih protokola; računala sa različitim OS-om mogu razmijenjivati podatke
- sigurnost – (ne)dozvoljavanje pristupa hardveru ili datotekama na računalu; autorizacija korisnika; unutarnja/vanjska sigurnost
- korisničko sučelje – command line, graphical user interface (GUI)

Binarno zapisivanje podataka

- prikaz podataka *brojevima*
 - ~ proizvoljni *brojevni sustav*
- ostvarivanje u *tehničkom* sustavu
 - ~ predočavanje znamenki posebnim fizičkim stanjem,
"na prikladan način"
- različita stanja:
 - jasno prepoznavanje
 - jasno međusobno razlikovanje
- najjednostavnije i najefikasnije ~ ostvarenje *2 stanja*
⇒ *binarni sustav* je osnova
svih digitalnih elektroničkih sustava

Binarno zapisivanje podataka

- binarna znamenka (0 ili 1) naziva se *bit* (engl. **binary digit**)
- grupiranje bitova u digitalnim sustavima radi prikaza podataka:
 - grupa od 8 bitova: *oktet* (engl. byte)
 - grupa od 4 bita: *kvartet* (engl. nibble)
 - osnovna grupa bitova: *riječ* (engl. word); tipično 8, 16, 32, 64, ... bita

Binarno zapisivanje podataka

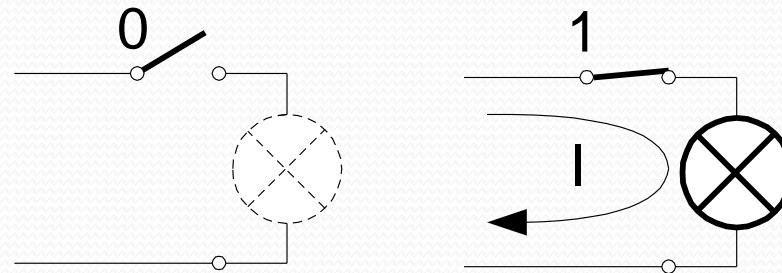
- veće grupiranje riječi ~ *blokovi*:
 - pohrana na magnetskim medijima (diskovi, trake)
~ koriste se elektromagnetski uređaji (*spori!*)
 - vrijeme pristupa usporedivo
s vremenom čitanja podataka
- efikasnost pristupa (čitanja/pisanja)

Ostvarenje binarnog zapisa

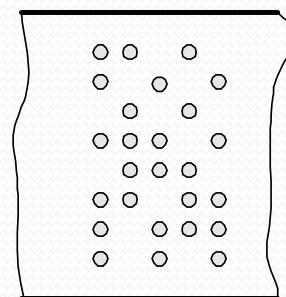
- binarne znamenke: 0 i 1
- fizičko predočavanje:
 - mehanička sklopka
 - papirna traka
 - magnetski medij
 - tranzistorska sklopka (elektronički sklop!)
- nositelj informacije:
 - pozitivni i negativni *impulsi* (struja ili napon) $\sim 0, 1$
 - *nizovi impulsa* \sim grupe 0, 1

Predočavanje binarnih veličina

- mehanička sklopka ("kontakt")

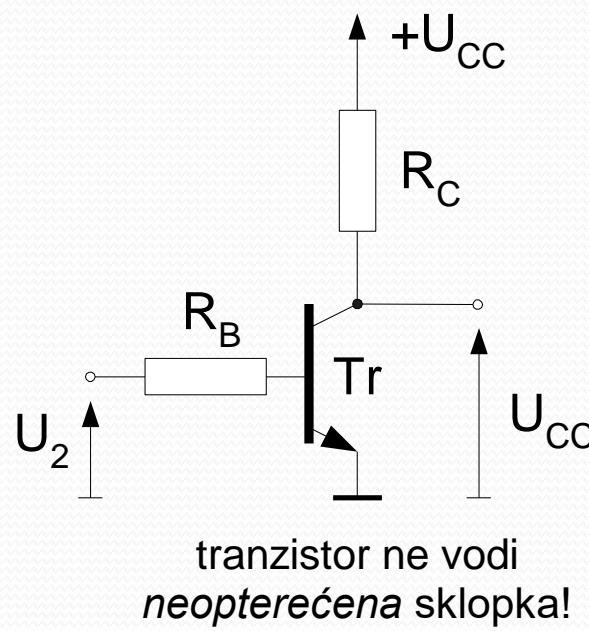
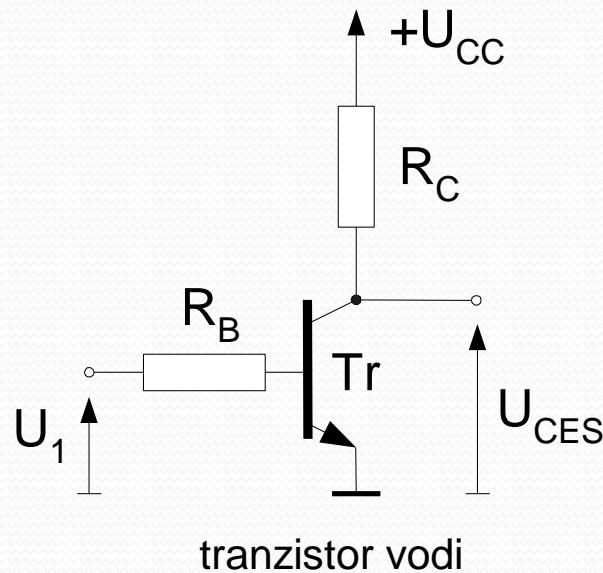


- bušena papirna traka/bušene papirne kartice



Predočavanje binarnih veličina

- primjer tranzistorske sklopke

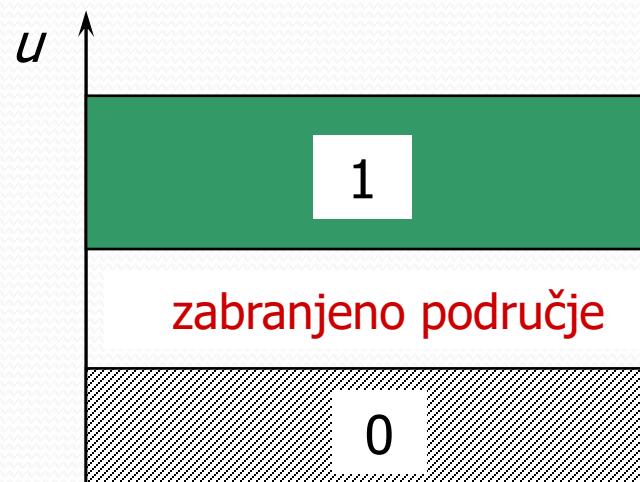


Predstavljanje naponskim razinama

- ostvarenje elektroničkim sklopovima:
o i 1 → naponske razine (N, V)

npr. o V (N) → "0", +5 V (V) → "1"

- problemi tehničke izvedbe
(tolerancije, opterećenja, otpornost na smetnje)
⇒ naponska *područja* umjesto razina



Prijenos binarnih podataka

- prijenos podataka (informacija)
 - ~ primanje i slanje:
 - *unutar* digitalnog sustava, između njegovih dijelova
 - *između* izdvojenih digitalnih sustava
- razlučiti trenutke očitanja vrijednosti pojedinog bita
⇒ *sinkronizacijski* (taktni) impulsi, CP (Clock Pulse)
- prijenos *binarnih* podataka (riječ, blok podataka):
 - serijski (po bitu), između digitalnih sustava, radi štednje
 - paralelno (po bitu), unutar digitalnog sustava, radi brzine

Tipovi i prikaz podataka

- zapis podataka (~ zapis bitovnog vektora):
utvrđeni oblik = *format*
 - organizacija niza bitova (grupe bitova ~ *polja*)
 - značenje pojedinih bitova/grupa bitova
- najjednostavniji zapis:
prirodni binarni brojevi
 - vrijednost bita u broju = pozicija bita u binarnom vektoru
- posve općenito:
pridruživanje značenja binarnom vektoru = *kôd*
 - broj
 - nešto drugo (~ simbol)

Pozicijski brojevni sustavi

- pozicija znamenke određuje njenu težinu
 - faktor kojim se znamenka množi
- težina - potencija *baze* brojevnog sustava
- dekadski sustav:
$$234 = 2 \cdot 10^2 + 3 \cdot 10^1 + 4 \cdot 10^0$$
- baza sustava može općenito biti bilo koji cijeli broj

Pozicijski brojevni sustavi

- prikaz n -znamenkastih *cijelih* brojeva:

$$N_B = a_{n-1} \cdot B^{n-1} + a_{n-2} \cdot B^{n-2} + \cdots + a_1 \cdot B^1 + a_0 \cdot B^0$$

$$= \sum_{i=0}^{n-1} a_i \cdot B^i$$

$$= a_{n-1}a_{n-2} \dots a_1a_0$$

B: *baza* ili *korijen* brojevnog sustava

a_i : koeficijent uz i -tu potenciju (težinu);

$$a_i = \{0, 1, \dots, B-1\}, i = 0, 1, \dots, n-1$$

~ znamenke

Prikaz razlomljenih brojeva

- princip prikaza kao za cijele brojeve:
težine znamenki iza zareza \sim negativne potencije baze

$$n_B = a_{-1} \cdot B^{-1} + a_{-2} \cdot B^{-2} + \cdots + a_{-(m-1)} \cdot B^{-m+1} + a_{-m} \cdot B^{-m}$$

$$= \sum_{i=-m}^{-1} a_i \cdot B^i$$

$$= 0, a_{-1} a_{-2} \dots a_{-(m-1)} a_{-m}$$

Miješani ili racionalni brojevi

- prikaz s *fiksnim zarezom* (engl. fixed-point notation) ~ "miješani" ili racionalni brojevi = cijeli broj + razlomljeni broj

$$N = N_B + n_B$$

$$= \sum_{i=-m}^{n-1} a_i \cdot B^i$$

$$= a_{n-1}a_{n-2}\dots a_1a_0, a_{-1}a_{-2}\dots a_{-(m-1)}a_{-m}$$

- pretvorba:
 - posebno cjelobrojni dio broja
 - posebno razlomljeni dio broja

Neki brojevni sustavi

baza B	brojevni sustav	znamenke sustava (B)
2	binarni	0,1
3	ternarni	0,1,2
8	oktalni	0,1,2,3,4,5,6,7
10	dekadski	0,1,2,3,4,5,6,7,8,9
16	heksadekadski	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

dekadski	binarni	oktalni	heksadekadski
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Pretvorba brojeva u različitim sustavima

- pretvorba *cijelog* dekadskog broja u neki drugi sustav
 - ~ uzastopno dijeljenje bazom tog sustava
 - ostaci dijeljenja s bazom ~ znamenke
 - ostatak prvog dijeljenja ~ najmanje značajna znamenka

Primjer:

$$N_{10} \rightarrow N_2 = b_{s-1}b_{s-2}\cdots b_1b_0$$

$$\begin{aligned}N_{10} &= b_{s-1} \cdot 2^{s-1} + b_{s-2} \cdot 2^{s-2} + \cdots + b_1 \cdot 2^1 + b_0 \cdot 2^0 \\&= 2 \cdot \left(b_{s-1} \cdot 2^{s-2} + b_{s-2} \cdot 2^{s-3} + \cdots + b_1 \cdot 2^0 \right) + b_0 \\&= 2 \cdot A_1 + b_0\end{aligned}$$

Pretvorba dekadskog broja u binarni

Primjer: $345_{10} \rightarrow ?_2$

$$345 : 2 = 172$$

$$172 : 2 = 86$$

$$86 : 2 = 43$$

$$43 : 2 = 21$$

$$21 : 2 = 10$$

$$10 : 2 = 5$$

$$5 : 2 = 2$$

$$2 : 2 = 1$$

$$1 : 2 = 0$$

1

0

0

1

1

0

1

0

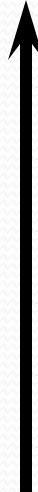
1



$$\Rightarrow 345_{10} = 101011001_2$$

Pretvorba dekadskog broja u ternarni

Primjer: $345_{10} \rightarrow ?_3$

$345 : 3 = 115$	0	
$115 : 3 = 38$	1	
$38 : 3 = 12$	2	
$12 : 3 = 4$	0	
$4 : 3 = 1$	1	
$1 : 3 = 0$	1	

$$\Rightarrow 345_{10} = 110210_3$$

Pretvorba dekadskog broja u heksadekadski

Primjer: $345_{10} \rightarrow ?_{16}$

$$345 : 16 = 21$$

$$21 : 16 = 1$$

$$1 : 16 = 0$$

9
5
1



$$\Rightarrow \quad 345_{10} = 159_{16}$$

Pretvorba binarnog broja u dekadski

- "direktna" pretvorba:
 - odrediti dekadski zapis težina (~ potencija baze) izvornog sustava
 - pomnožiti vrijednost svake znamenke s odgovarajućom težinom
 - sumirati

Primjer: $10010_2 \rightarrow ?_{10}$

$$\begin{aligned}10010_2 &= 1*2^4 + 0*2^3 + 0*2^2 + 1*2^1 + 0*2^0 \\&= 1*16 + 1*2 = 18\end{aligned}$$

$$\Rightarrow 10010_2 = 18_{10}$$

Oktalni i heksadekadski sustav

Primjer : $101111011001100_2 \rightarrow ?_8$

101	111	011	001	100
5	7	3	1	4

$$101111011001100_2 = 57314_8$$

Primjer : $765432_8 \rightarrow ?_2$

7	6	5	4	3	2
111	110	101	100	011	010
$765432_8 = 111110101100011010_2$					

Heksadekadski sustav

Primjer : $0101110001110011100_2 \rightarrow ?_{16}$

0101	1110	0011	1001	1100
5	E	3	9	C

$$0101110001110011100_2 = 5E39C_{16}$$

Primjer : $76A4C2_{16} \rightarrow ?_2$

7	6	A	4	C	2
0111	0110	1010	0100	1100	0010

$$76A4C2_{16} = 011101101010010011000010_2$$

Logika sudova i računalni sustav

- *digitalna logika* : principi funkcioniranja i načini izvedbe digitalnih sklopova i sustava
- digitalni (računalni) sustav
 - ~ sve funkcije temeljene na malom skupu "osnovnih logičkih funkcija"
- sklopovi koji ostvaruju osnovne logičke funkcije
 - ~ osnovni logički sklopovi:
obrađuju "logičke varijable"



Logika sudova i računalni sustav

- elektroničke izvedbe osnovnih logičkih sklopova:
"Električke veličine koje odgovaraju logičkim varijablama održavaju se unutar unaprijed definiranih i fiksnih granica (na ulazu i na izlazu)."
- "logičke varijable", "osnovne logičke funkcije"
 - ~ terminologija logike sudova
- *logika sudova, propozicijska logika*
(engl. propositional logic)
 - ~ "kombiniranje" *elementarnih* sudova radi dobivanja novih *složenih* sudova, bez obzira na suvislost samih sudova
- osnovni *kombinatori* sudova
 - ~ "osnovni logički veznici"

Logika sudova i i računalni sustav

- sudovi (tvrdnje, iskazi):
 - jednostavne rečenice
 - istiniti ili neistiniti

Primjer:

sud A: "Nema ulja (u motoru)."

sud B: "Temperatura (motora) je previsoka."

Logički kombinatori

- osnovni logički veznici:
 - ~ "kombinatori" I, ILI
- vrijednost složenog suda
 - ~ istinit ili neistinit

Primjer:

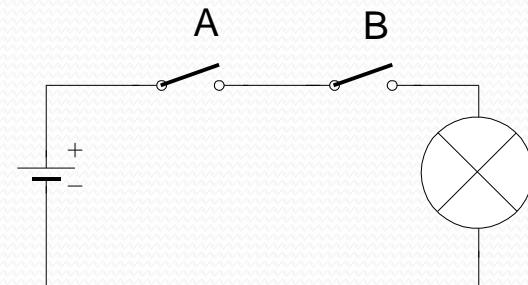
$f = A \text{ ILI } B$ = "Nema ulja (u motoru)."
 ILI "Temperatura (motora) je previsoka."

$f = A \text{ I } B$ = "Nema ulja (u motoru)."
 I "Temperatura (motora) je previsoka."

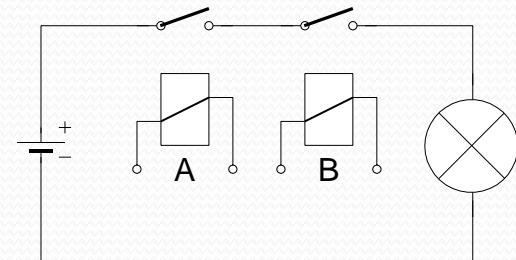
Logički kombinatori

- izvedba kombinatora I
 - (mehanički) kontakt:

A ≡ <sklopka A uključena>
B ≡ <sklopka B uključena>
f ≡ <žarulja svjetli>

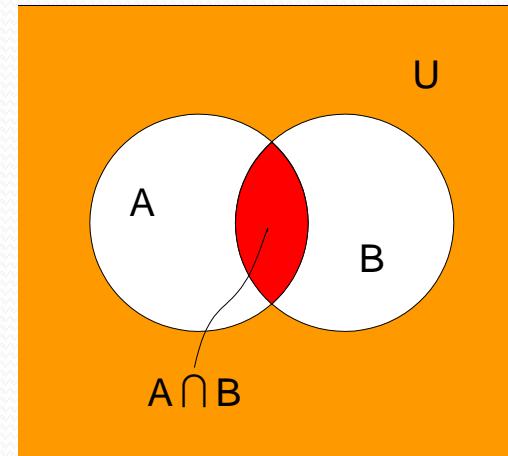


- izvedba relejima:
struja = pobuda releja



Interpretacija kombiniranja

- algoritamski:
ako (A istinit) **i** (B istinit)
onda f istinit
inače f neistinit
- "logički produkt"
~ *konjunkcija*
 - "računarska" notacija:
 - simbolička logika:
 - teorija skupova:



$$f = A \cdot B = AB$$

$$f = A \wedge B$$

$$f = A \cap B$$

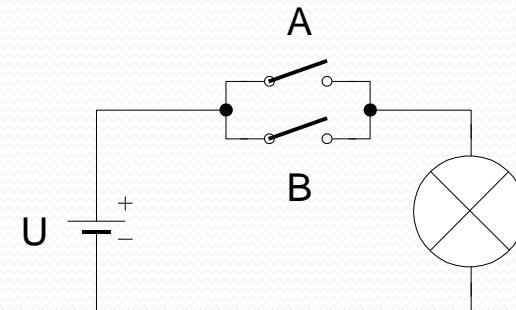
Logički kombinatori

- izvedba kombinatora ILI
 - (mehanički) kontakt:

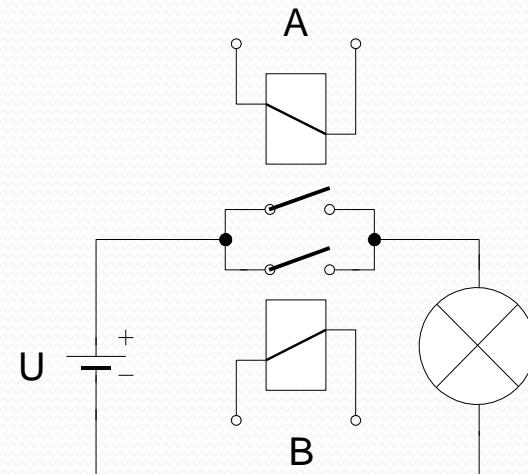
A ≡ <sklopka A uključena>

B ≡ <sklopka B uključena>

f ≡ <žarulja svijetli>

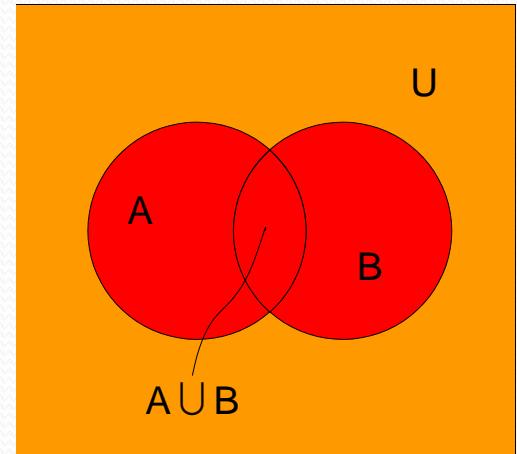


- izvedba relejima:
struja = pobuda releja



Interpretacija kombiniranja

- algoritamski
ako (A istinit) *ili* (B istinit) (*ili oba!*)
onda f istinit
inače f neistinit



- "logička suma"
~ disjunkcija $f = A + B$
- "računarska" notacija: $f = A \vee B$
- simbolička logika: $f = A \cup B$
- teorija skupova:

Tablice istinitosti (kombinacija)

- *tablica kombinacija, tablica istinitosti* (engl. truth table)
 - ~ prikaz djelovanja kombinatora:
 - konačni broj mogućih kombinacija
 - vrijednosti istinitosti elementarnih sudova
- oznake: T ~ istina, \perp ~ neistina
- definiraju odnos ulaza i izlaza digitalnog sustava

funkcija I
(konjunkcija)

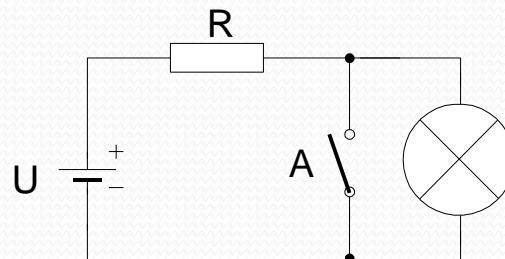
A	B	f
\perp	\perp	\perp
\perp	T	\perp
T	\perp	\perp
T	T	T

funkcija ILI
(inkluzivna disjunkcija)

A	B	f
\perp	\perp	\perp
\perp	T	T
T	\perp	T
T	T	T

Logička negacija

- logička funkcija *NE*, komplement, inverzija
- nije kombinator (ali je korisni operator ☺)
- algoritamski
 - ako (A istinit)
onda f neistinit
inače f istinit
- logički izraz
 - "računarska" notacija:
 - simbolička logika:
 - teorija skupova:



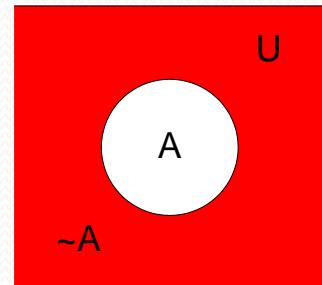
funkcija NE
(negacija)

$$f = \bar{A}$$

$$f = \neg A$$

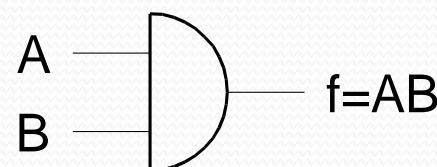
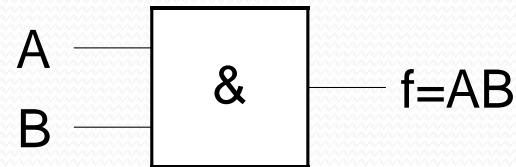
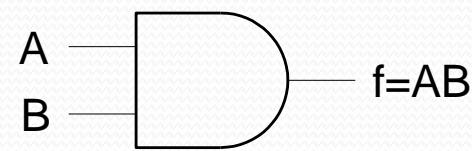
$$f = A^C$$

A	f
⊥	T
T	⊥



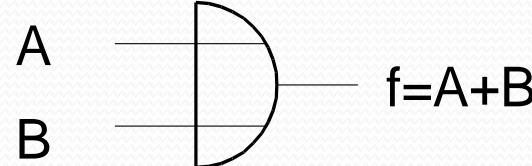
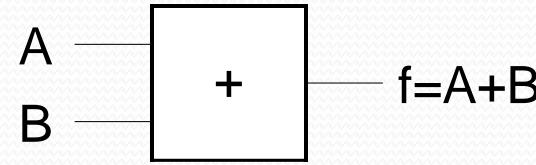
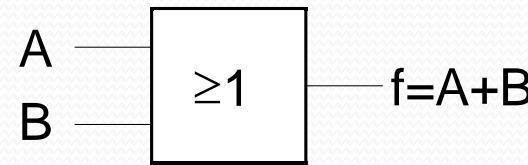
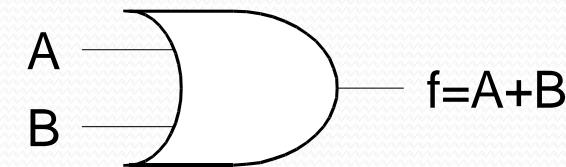
Simboli za logičke kombinatore

- simboli za kombinator I:
 - američki vojni standard Mil-STD-806B
 - međunarodni standard IEC/ISO,
DIN 40900,
ANSI/IEEE 91-1984
 - stari standard DIN



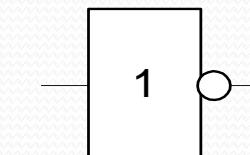
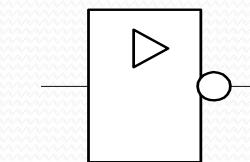
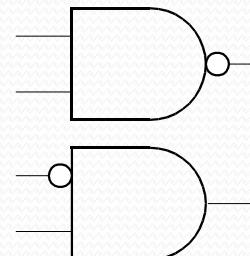
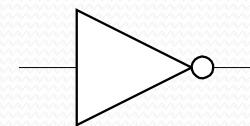
Simboli za logičke kombinatore

- simboli za kombinator ILI:
 - američki vojni standard Mil-STD-806B
 - međunarodni standard IEC/ISO,
DIN 40900,
ANSI/IEEE 91-1984
 - stari standard DIN



Simbol za logičku negaciju

- simboli za operator NE:
 - američki vojni standard Mil-STD-806B
 - kombiniranje s drugim operatorima
 - međunarodni standard IEC/ISO



Softver I

- softver - skup (strojnih) instrukcija temeljem kojih procesor računala izvršava određene zadaće
- neopipljivi dio računalnog sustava
- softver: programi, biblioteke funkcija (podataka), dokumentacija...
- instrukcije – (ne)vidljivi rezultat

Softver II

- viši programski jezik
- prevoditelj (kompajler, interpreter)
- strojni kod
- izvorni kod (source code)

računalni program u razumljivoj formi čovjeku

```
1  sub Calculator()
2  sub addition(self, other)
3      return self + other
4  end
5
6  /** Create a list of 2 numbers. */
7  sub makeFraction(numerator, denominator)
8      return [numerator, denominator]
9  end
10
11 /**
12  * Warning: Destroys original fraction!
13  */
14 sub multiplyFracs(frac, otherFrac)
15     frac[0] *= otherFrac[0]
16     frac[1] *= otherFrac[1]
17     return frac
18 end
19
20 sub InfinityCalculator()
21     inherit Calculator()
22     /** Create a list of 2 numbers. */
23     sub makeFraction(numerator, denominator)
24         if denominator == 0
25             /* The user is trying to divide by 0.
26             * Use Java's way of handling this: */
27             import math into mathematics
28             return mathematics.INFINITY
29         end
30     end
31 end
```

Method

Class

Podjela

- domena korištenja koda:
 - korisnički softver
 - sistemski softver: operacijski sustav, pogonski programi (drivers), programi za održavanje računala (antivirus)
 - virusni (malware) softver
- mjesto izvršenja koda:
 - desktop aplikacije
 - server aplikacije
 - ugradbeni softver
 - plugin-ovi
- dozvola korištenja (licenca):
 - freeware softver (funkcionalnost?)
 - komercijalni softver

Softverska obilježja

- razvojni alat za pisanje softvera (Eclipse, Microsoft Visual Studio): compiler, debugger, linker, text editor
- lanac izvođenja: korisnički softver – sistemski softver – strojni kod (hardver)
- problemi i pogreške u izvođenju – buggs
- procedure za testiranje softvera
- patentna zaštita softvera: "patent on any performance of a computer realized by means of a computer program,"
 - ne podrazumijeva samo ideju „što bi softver mogao raditi“
 - konkretna softverska implementacija obično nije patentibilna ali podliježe zaštiti autorskih prava
- otvorenost koda

Otvoreno računarstvo I

- otvoreni zapis raznih formata (OpenOffice.org, Google.docs)
 - mogućnost da svatko može napraviti program za čitanje/upis podataka
- otvoreni protokoli za komunikaciju
 - slobodna razmjena podataka između jedinki sustava TCP (Transmission Control Protocol)
- otvoreno sklopoljje
 - lagana dogradnja/izmjena pojedinih hardverskih komponenti
- otvorena programska podrška
 - mogućnost dorade programske podrške, uvid u sami kod

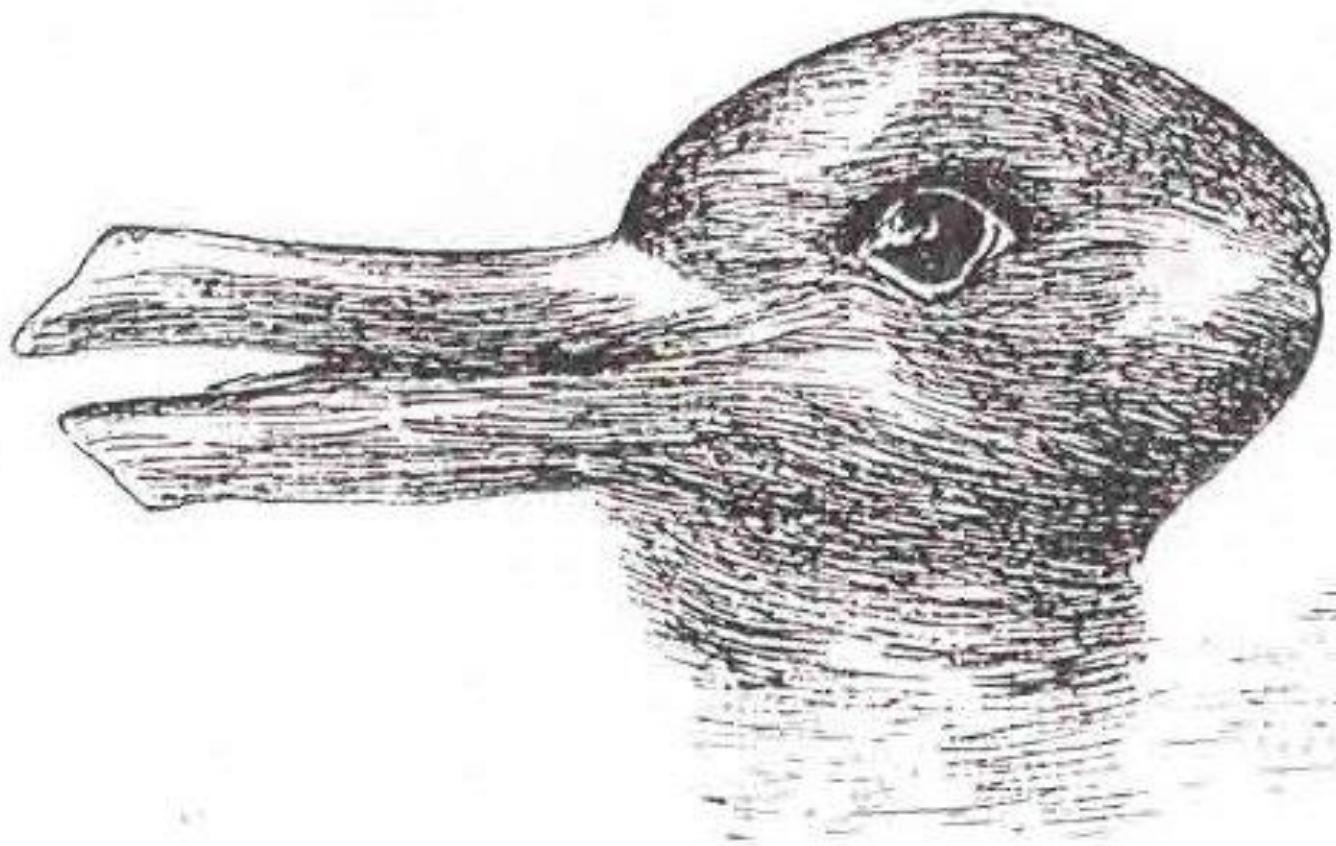
Otvoreno računarstvo II

- “Otvoreni sustav je onaj koji sadrži dovoljan broj otvorenih specifikacija za sučelja, servise i formate podataka, kako bi se aplikacijama omogućilo:
 - da uz što manje promjena budu prenosive na druge računalne sustave (*portability*)
 - da uspješno surađuju s drugim aplikacijama na vlastitim i udaljenim sustavima (*interoperability*)
 - komuniciranje s korisnikom na njemu ugodan način uz lako privikavanje na rad s različitim sustavima (*user portability*).“

Otvoreno računarstvo III

- nedostatci
 - problem sigurnosti (razmjena ideja, podataka...)
 - nisu sve otvorene norme nužno (dovoljno) dobre
 - prijelaz zatvoreno=>otvoreno nosi dodatne financijske i organizacijske probleme
- na početku (60-tih) programeri slobodno izmjenjuju programe ideje itd.
- pojavom računala u široj uporabi (70-tih), pojedine kompanije žele kontrolu nad izvornim kodom, normama itd.
- 80-te, javljaju se prve grupe/udruženja koja se opiru zatvorenosti u računarstvu (1988. Open Software Foundation (OSF))
- politika uglavnom (barem deklarativno) podržava otvoreno računarstvo

Otvorenost sagledavanju i rješavanju problema



Literatura

- S. Ribarić. Građa računala – arhitektura i organizacija računarskih sustava.



- U. Peruško, V. Glavinić: *Digitalni sustavi*

