

# Uvod

# Kurs iz Uvoda u programiranje

- Predmetni nastavnik:  
Miloš Radovanović
  
- Asistenti:  
Saša Tošić  
Miloš Savić  
Davorka Radaković  
Tijana Vislavski
  
- E-mail: ime.prezime@dmi.uns.ac.rs (bez nasih slova)
  - Osim Tijane (za sad): tijana.pzs@gmail.com

# Kurs iz Uvoda u programiranje

- Oblici nastave:
  - Predavanja (Radovanović)
  - Vežbe (Radovanović RN, Tošić IT, po rasporedu)
  - **Praktične vežbe (Savić, Radaković, Vislavski), petkom**
  
- Sajt kursa: **[perun.pmf.uns.ac.rs/moodle/](http://perun.pmf.uns.ac.rs/moodle/)**
  - Napravite korisnički nalog
  - Prijavite se na kurs (enrolment key: Java2015)
  - Istaknut raspored grupa za praktične vežbe
    - Praktične vežbe počinju 16. 10.
    - Kontakt osoba za raspored: Davorka Radaković

# Kurs iz Uvoda u programiranje

- Materijal sa predavanja i vežbi biće redovno postavljan na sajt kursa
- Ne postoji jedinstven udžbenik, ali sledeća literatura može pomoći:
  - **Programski jezik Java:** Mirjana Ivanović, Mihal Bađonski, Zoran Budimac, Dragoslav Pešović. Departman za matematiku i informatiku, Prirodno-matematički fakultet, Novi Sad, 2005.
    - Uskoro se očekuje izlazak novog izdanja
  - **Uvod u programiranje i programski jezik Modula-2:** Zoran Budimac, Mirjana Ivanović, Đura Paunić. Departman za matematiku i informatiku, Prirodno-matematički fakultet, Novi Sad, 1998, 2004, 2011.
    - Pokriva deo kursa o iterativnim i rekurzivnim postupcima i dr. ali u programskom jeziku Modula-2
  - **Zbirka zadataka iz Uvoda u programiranje:** Mirjana Ivanović, Mihal Bađonski, Aleksandar Popović, Miloš Radovanović. Feljton & Departman za matematiku i informatiku, Prirodno-matematički fakultet, Novi Sad, 2003.
    - Pokriva deo zadataka sa vežbi, ali u programskom jeziku Modula-2

# Programiranje

- Pod (računarskim) **programiranjem** obično se podrazumeva čovekova aktivnost (odnosno, aktivnost **programera**) usmerena ka rešenju nekog konkretnog problema na računaru
- Problem se obično rešava konstrukcijom **algoritma** za rešenje problema, a algoritam se zapisuje posredstvom **programskog jezika**
- Po jednoj od najopštijih definicija programskog jezika njegova uloga je da obezbedi konstrukcije (i načine) za organizovanje “izračunavanja” na računaru
- Tako organizovano izračunavanje nazivamo program izračunavanja, odnosno kraće: **program**

# Programi

- Kao sastavni deo rešenja problema programi obično sadrže interakciju sa korisnikom, kao i drugim računarima i uređajima, jednim terminom zvani **ulazno-izlazne operacije**
- Interakcija sa korisnikom obavlja se kroz **korisnički interfejs**
  - Konzolni (*console, command prompt*)
  - Grafički (*graphical user interface* – GUI)
- Komunikacija sa drugim računarima i uređajima uključuje:
  - Rad sa fajlovima
  - Razmenu informacija preko mreže
  - Štampanje
  - ...

# Programi

- Razlikujemo programe **nižeg i višeg nivoa** (tačnije, hijerarhiju nivoa gde se viši nivoi oslanjaju na usluge nižih)
- U programe nižeg nivoa spadaju operativni sistemi (Windows, Linux, Android...) i njihovi delovi, drajveri, itd.
  - Programi “bliži” hardveru računara
  - Operativni sistemi se obrađuju na istoimenim kursevima (OS 1 i 2)
- U programe višeg nivoa spadaju korisnički programi (aplikacije), npr. Word, Skype, Grand Theft Auto...
  - Programi “dalji” i nezavisniji od hardvera
  - Na ovom kursu se fokusiramo na ovu vrstu programa, sa konzolnim korisničkim interfejsom
  - GUI će se raditi na drugim kursevima (Objektno-orijentisano programiranje 1 i 2, Web dizajn...)

# Računari

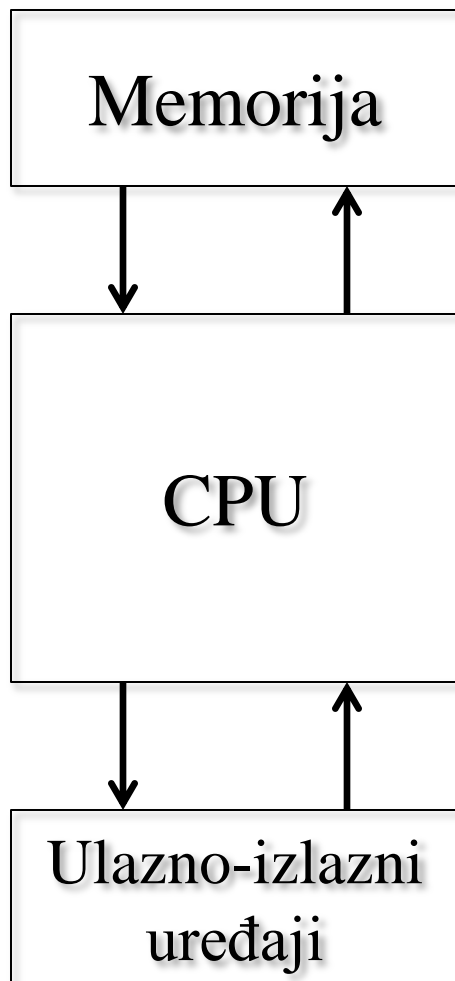
- Pojam **računar** (*computer*) obuhvata sve uređaje koji se sastoje od tri osnovne komponente
  - Centralni procesor (*central processing unit* – CPU)
  - Memorija
  - Mogućnost ulaza/izlaza (*input/output* – IO)
- Uz to postoje dodatni zahtevi koji se tiču mogućnosti procesora u smislu izračunljivosti (kursevi: Teorijske osnove informatike 2, Analiza algoritama)
- **Računarska platforma**: svaki uređaj koji ima gore navedene elemente i poseduje operativni sistem koji omogućava pravljenje i instalaciju korisničkih aplikacija širokih mogućnosti



# Računari

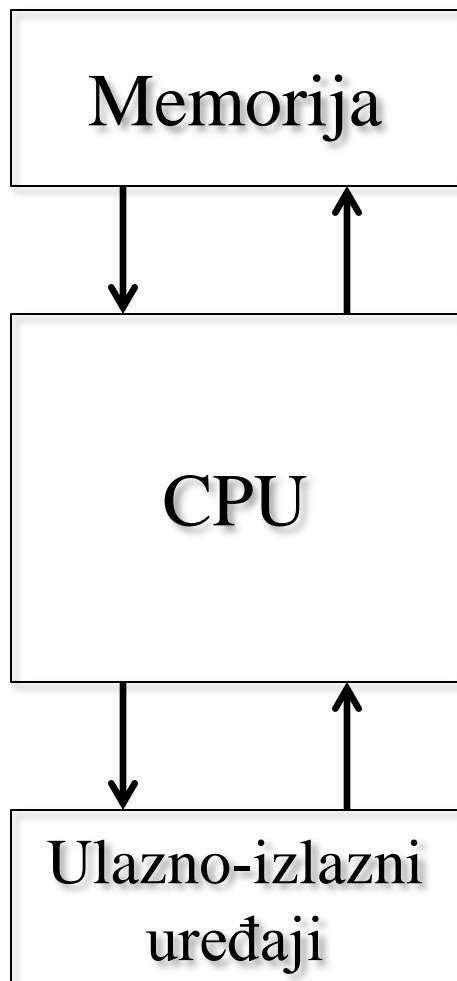
- Primeri računara:
  - Personalni (desktop) i notebook računari
  - Tableti
  - “Pametni” telefoni i televizori
- Nisu računari:
  - Klasični kalkulatori
  - Stari telefoni i televizori
- “Na granici” (jesu računari, ali ne i platforme):
  - Ruteri
  - Specijalizovani industrijski kontroleri

# Računari i programi



- Osnovna podela sadržaja memorije i IO uređaja:
  - Programski kod (instrukcije) u mašinskom jeziku
  - Podaci
- U osnovi, SVI sadržaji su predstavljeni binarnim brojevima (nizovima 0 i 1)
- Program (na primer, .exe fajl) se učitava sa ulaznog uređaja, CPU ga izvršava koristeći memoriju, učitavajući podatke sa ulaznih uređaja i šaljući rezultat(e) na izlaz

# Računari i programi



- Jedan prost (hipotetički) mašinski program od 3 instrukcije:

```

00000100111100111010001010
00000101000100100111100010
00101100101010111111001111
  
```

- Isti program u (hipotetičkom) asemblerskom jeziku:

**LOAD**     A

**ADD**       B

**STORE**    C

- Program odgovara naredbi dodele  $A = B + C$ ,  
gde su A, B i C imena memorijskih lokacija

# Programski jezici

- **Mašinski jezik** je jedini jezik koji računar “razume”, tj. programe napisane u mašinskom jeziku jedine direktno može da izvršava
- Prvi programeri (1950-ih) su pisali programe u mašinskom jeziku
- **Nedostaci programiranja u mašinskom jeziku** su brojni:
  - Užasno je teško pisati programe
  - Takođe je teško razumeti ih, nalaziti greške, i menjati
    - Programerima teško da čitaju sopstveni programski kod nakon izvesnog vremena, kamoli tuđ
  - Svaki model računara ima drugačiji mašinski jezik
    - Da bi se program mogao izvršavati na drugom modelu računara, treba ga u osnovi ponovo napisati

# Programski jezici

- Ubrzo su se pojavili **asembler**ski jezici
  - Pojedinačne binarne instrukcije, pa i grupe instrukcija, zamenjene simboličkim, a programi se pišu u formi teksta
  - Poseban program, **asembler** (*assembler*) zadužen je za prevođenje programa pisanog u assemblerkom jeziku u mašinski jezik
  - Programiranje donekle olakšano
  - I dalje različiti modeli računara imaju različite asemblerke jezike
- Ovaj “niski nivo” računara i programiranja obrađuje se na kursu Organizacija računara

# Programski jezici

- Asemblerski jezici su pokrenuli proces **apstrakcije** programa i programiranja od arhitekture konkretnog računara
- **Viši programski jezici** predstavljaju sledeći korak u tom smeru
- Konstrukcije viših jezika su u potpunosti (ili bar u velikoj meri) nezavisne od računara, a kao i kod asemblerskih jezika programi su u formi teksta
  - Npr. naredba dodele  $A = B + C$  će u memorijsku lokaciju (promenljivu) sa imenom  $A$  smestiti broj dobijen sabiranjem brojeva sadržanih u memorijskim lokacijama sa imenima  $B$  i  $C$
- Poseban program, **kompajler** (*compiler*) zadužen je za prevođenje programa pisanog u višem programskom jeziku u mašinski jezik
- Broj viših programskih jezika u aktivnoj upotrebi meri se u hiljadama
- Primeri viših programskih jezika: FORTRAN, C, C++, BASIC, Pascal, Modula-2, LISP, Smalltalk, Python, **Java**...

# Java

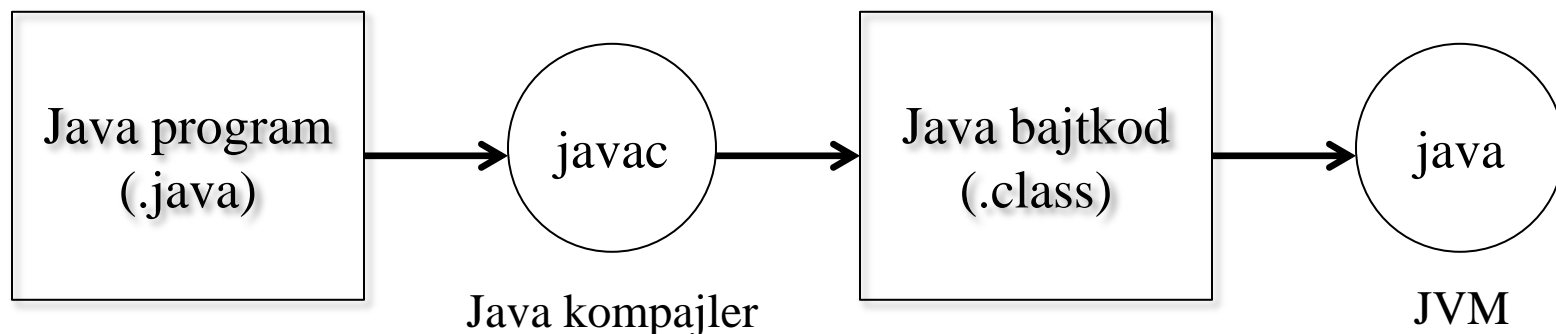
- **Java** je veoma popularan i rasprostranjen viši programski jezik, zvanično predstavljen javnosti 23. maja 1995. godine
- Razvio ga je *Sun Microsystems* u okviru projekta *Green*, kao jezik i platformu za kontrolisanje rada potrošačkih elektronskih uređaja
- U Sun-u su uskoro uvideli da je novi jezik pogodan i za Web
- Glavni prodor je napravljen kada je *Netscape* dobio licencu na Javu i u septembru 1995. godine predstavio svoj novi brauzer *Netscape Navigator 2.0* sa podrškom za Java aplete
- Korporaciju Sun je u međuvremenu preuzeo softverski gigant *Oracle* (poznat po sistemima za upravljanje bazama podataka)
- Rasprostranjenosti Jave veoma doprinosi Android operativni sistem, gde su programi (*app*-ovi) bazirani na Javi

# JVM

- Kao i kod drugih viših programskih jezika, Java kod se prevodi u mašinski jezik, i to poseban jezik nazvan Java **bajtkod** (*bytecode*)
- Međutim, ne postoji fizički računar koji “razume” Java bajtkod, već se radi o “virtuelnom” računaru koji se zove **Javina virtuelna mašina** (*Java Virtual Machine* – JVM)
- Fizički računar zamenjuje program koji ga simulira (rad CPU, memorije, ulaz/izlaz) i naziva se **JVM interpreter**, ili prosto JVM
- Implementacije JVM postoje za sve značajnije računarske platforme
- **Prednost virtuelnog pristupa:** Java programi kompajlirani u bajtkod na jednoj platformi (trebalo bi da) se mogu izvršavati bez izmena na svakoj platformi za koju postoji implementacija JVM
- **Mana virtuelnog pristupa:** manja efikasnost programa
- Različiti aspekti kompajliranja, interpretiranja, virtuelnih mašina itd. obrađuju se na kursevima Programski jezici i Konstrukcija kompajlera



# Izvršavanje Java programa



- U dijagramu, "javac" i "java" su konzolni programi koji dolaze u sastavu Java Development Kit (JDK)
- Moderna okruženja za razvoj programa (*integrated development environment* – IDE) obično sakrivaju ove korake pod jednim dugmetom "Run", ali oni su i dalje prisutni