# Zgodovina

## Kaj je programski jezik

**Univerazlni jezik,** je jezik v katerem je možno načeloma rešiti poljuben problem, ki se ga sploh da rešiti na računalniku.

**Neuniverzalni jezik,** je jezik v katerem lahko učinkovito rešimo tipične probleme v dani problemski domeni.

* Sintaksa programskega jezika – enolično določa obliko
* Semantika programskega jezika – enolično določa interpretacijo (pomen)
* Pragmatika programskega jezika – govori o izvoru in razvoju jezika

## Prevajanje programskih jezikov

* Pravajalnik
* Interpreter
* Narava jezika, ki ga prevajalnik prevaja
* Namen, ki ga s prevajalnikom želimo doseči

## Razširjenost programskih jezikov

* Preprost za učenje in programirane
* Usmerjenost velikih korporacij v uporabu jezikov
* Razpoložljivost prevajalnikov
* Potrebe po programski opremi
* Lastnosti programskega jezika
* Standardizacija jezika

## Razvoj programskih jezikov

* Razvoj tehnologije
* Dolgoletne izkušnje pri uporabi obstoječih programskih jezikov
* Potrebe po programski opremi
1. Strojni jezik
2. Zbirni jezik
3. Višjenivojski postopkovni jezik – imperativni (ukazni) jeziki
4. Nepostopkovni sprecializirani jeziki

# Lastnosti programskih jezikov

## Kriteriji za načrtovanje jezikov

* Varnost
* Berljivost
* Prožnost
* Dinamičnost
* Univerzalnost
* Preprostost in prenosljivost
* Učinkovitost

## Kriteriji za ocenjevanje jezikov

* Primernost
* Razpoložjivost programerjev
* Kvaliteto prevajalnikov in dokumentacije
* Različne vidike učinkovitosti

## Škodljive lastnosti programskih jezikov

* Stavek GOTO
* Dvoumnost sintakse
* Implicitna in dinamična deklaracija imen
* Šibko tipiziranje
* Stranski učinki
* Več imen za en objekt
* En konstrukt z več pomeni
* Dinamično spreminjanje programa

# Tipiziranje

Tip podatkovnega objekta definira:

* Strukturo objekta
* Množico možnih vrednosti
* Možne operacije na objektu

Jakost tipiziranja:

* Zbirnik – programski jezik brez tipiziranja
* Implicitno tipiziranje
* Dinamično tipiziranje
* Eksplicitno tipiziranje
	+ Šibko tipiziranje
	+ Močno tipiziranje po strukturi
	+ Močno tipiziranje po imenu
	+ Objekto tipiziranje

Novi tip je lahko definiran (eksplicitno tipiziranje):

* S podmnožico vrednosti obstoječega tipa
* S strukturo, sestavljeno iz obstoječih tipov
* Z množico novih vrednosti

# Sočasnost

* Za opisovanje algoritmov, ki so naravno predstavljeni z več procesi, ki se izvajajo sočasno
* Za pohitritev izvajanja na računalniku z dvema ali več procesorskkimi enotami
1. semaforji
2. Monitor
3. Srečanje
	1. Simetrično srečanje
	2. Asimetrično srečanje