Algoritmi in programiranje

III. gimnazija Maribor

Informatika

Problem

- Na začetku imamo problem, ki ga želimo rešiti s pomočjo računalnika.
- Če program za naš problem že obstaja, ga kar uporabimo, če pa takšnega programa še ni, pa se lahko sami lotimo programiranja

1. korak k rešitvi problema je: OPREDELITEV PROBLEMA

- S tem se ukvarja posebno področje računalništva, ki se imenuje INFORMACIJSKA SISTEMSKA ANALIZA.
- Bolje je problem opredeljen, lažja je pot do rešitve.

2. korak k rešitvi problema je: ZAPIS NAVODIL ZA REŠITEV PROBLEMA - ALGORITEM

- V algoritmu opredelimo postopek reševanja problema in določimo operacije, ki jih bomo pri tem uporabili.
- Vsak algoritem se mora enkrat končati.
- Algoritem je zaporedje navodil, ki v končnem številu korakov privede do rezultata.

ALGORITEM

- Kako podroben je algoritem je odvisno od tega, za koga ga pišemo.
- Pri razvijanju algoritmov za računalniško reševanje problemov izberemo postopek, ki ustreza temu načinu reševanja.
- Izdelaj:
  - algoritem za peko palačink,
  - algoritem za izdelavo vabil za rojstni dan z urejevalnikom besedil.

Potrebujemo recept in postopek izdelave..

1. korak

3 jajca
25 dag moke
4 dcl mleka
4 žlice olja
ščep soli

Umešamo testo.

2. korak

ponev

olje

Spečemo.

3. korak

Marelična marmelada

Namažemo in zvijemo.

DIAGRAM POTEKA

- Za nazornejši način predstavljamo algoritme z diagrami poteka.
- Elementi diagrama poteka so izbrani tako, da iz njih sestavljeni diagram poteka predstavlja primerno osnovo za kasnejše pisanje programa za računalnik.

SPREMENLJIVKE

- So poseben element diagrama poteka. Označevali jih bomo s simboli, kot pri matematiki: a, b, x, y,…
- Spremenljivkam lahko določimo ali priredimo vrednost in to simbolično zapišemo:

$$x \longleftarrow 5$$

Znak $\longleftarrow$ imenujemo prireditveni operator.

Prireditveni operator

Spremenljivka x naj dobi vrednost 10.

X ← 10

Spremenljivki n naj se vrednost poveča za 1 ali spremenljivka n naj dobi vrednost n+1.

n ← n+1

Spremenljivki A priredimo matematični izraz B+C.

A ← B+C

- Levi del prireditve je vedno spremenljivka, ki ji s prireditvijo določamo vrednost, desno pa je poljuben matematični izraz.

Opis elementov diagrama poteka

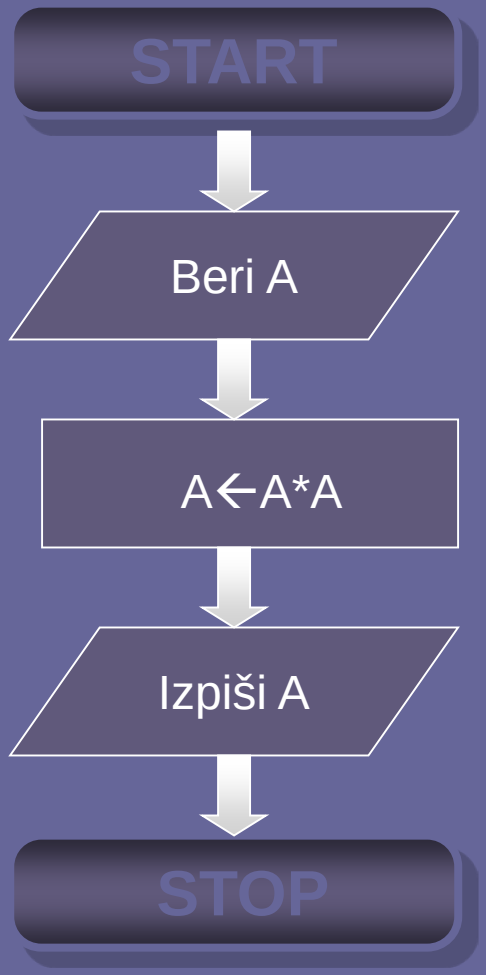
ZAČETEK ALGORITMA

KONEC ALGORITMA

PRIREDITEV VREDNOSTI

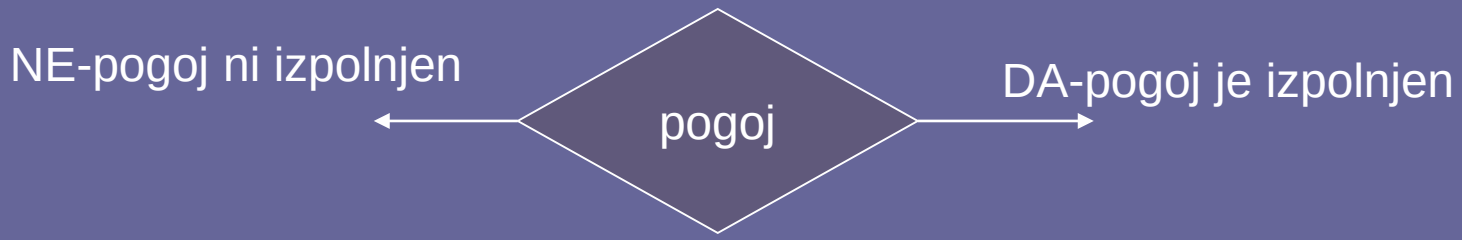
BRANJE IN IZPIS

Diagram poteka za KVADRIRANJE POLJUBNEGA ŠTEVILA



VEJITEV

- Pri sestavljanju algoritmov naletimo na položaj, ko moramo izvesti določeno akcijo glede na okoliščine.
- Del kjer se v algoritmu izvajanje cepi v dve veji, imenujemo vejitev.
- V diagramu poteka prikažemo vejitev z deltoidom ali ODLOČITVENIM BLOKOM:



VEJITEV

- Primer:

  Če dežuje, ostanemo doma, če ne, gremo na izlet na Pohorje. Akcijo izlet izvedemo le, če je izpolnjen določen pogoj.

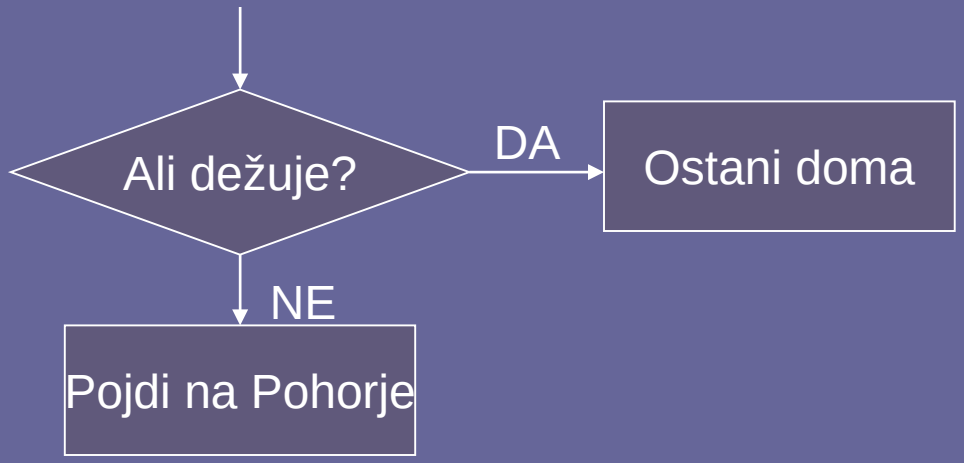
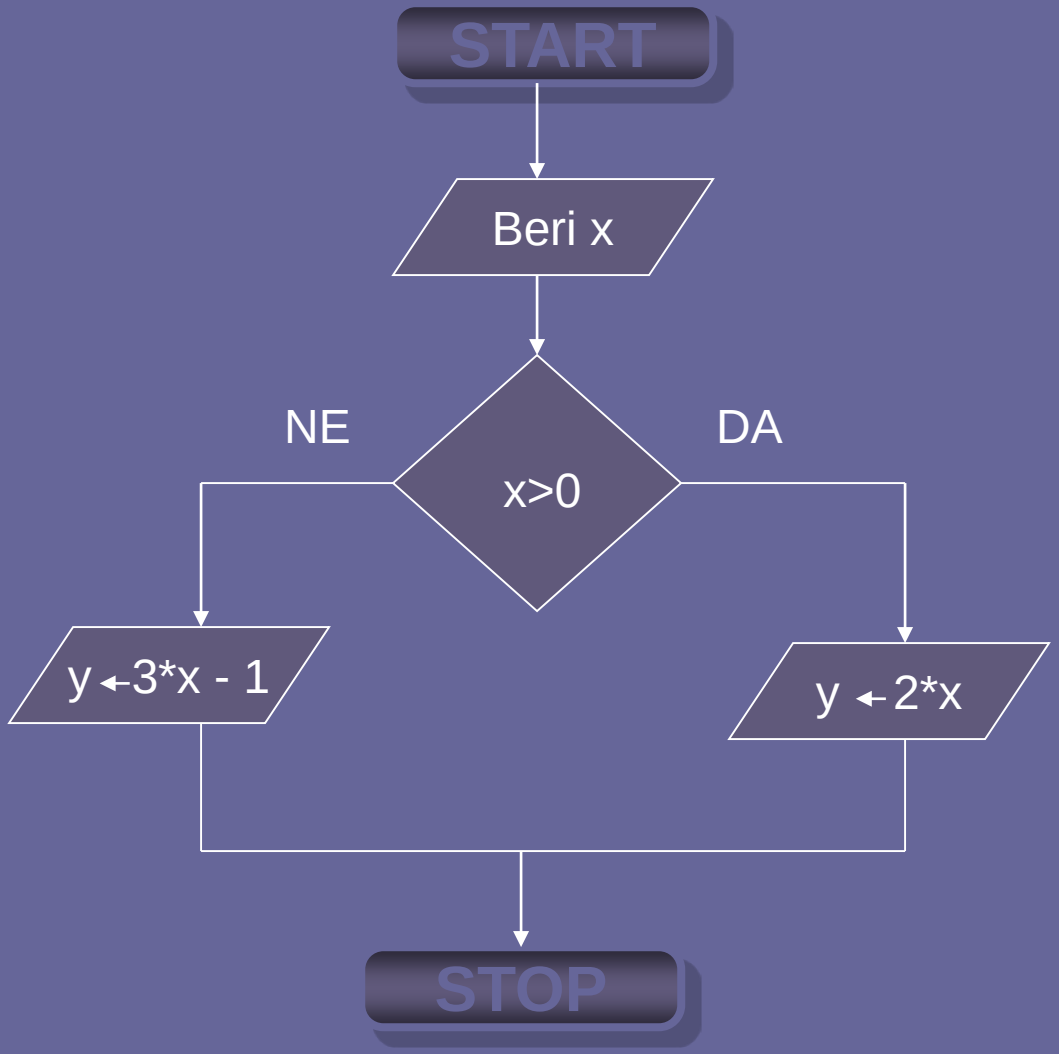


Diagram poteka z vejitvijo



Zanka

- Za zaporedje enakih ukazov uporabimo v diagramih poteka zanko.

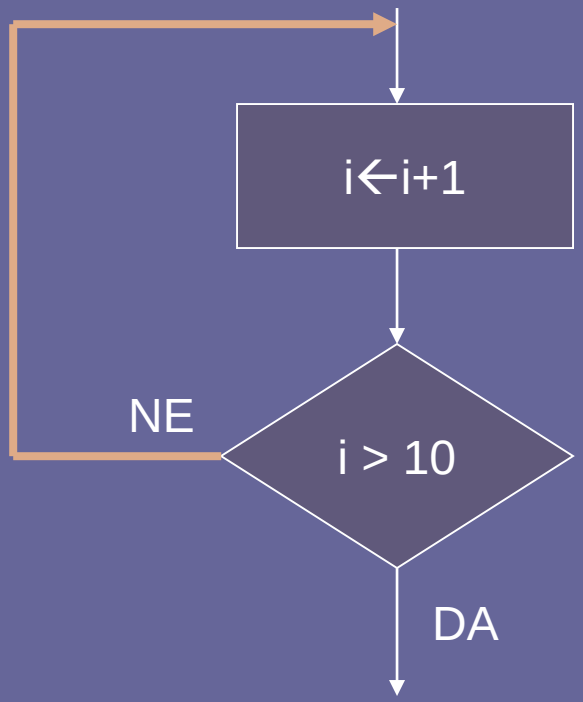
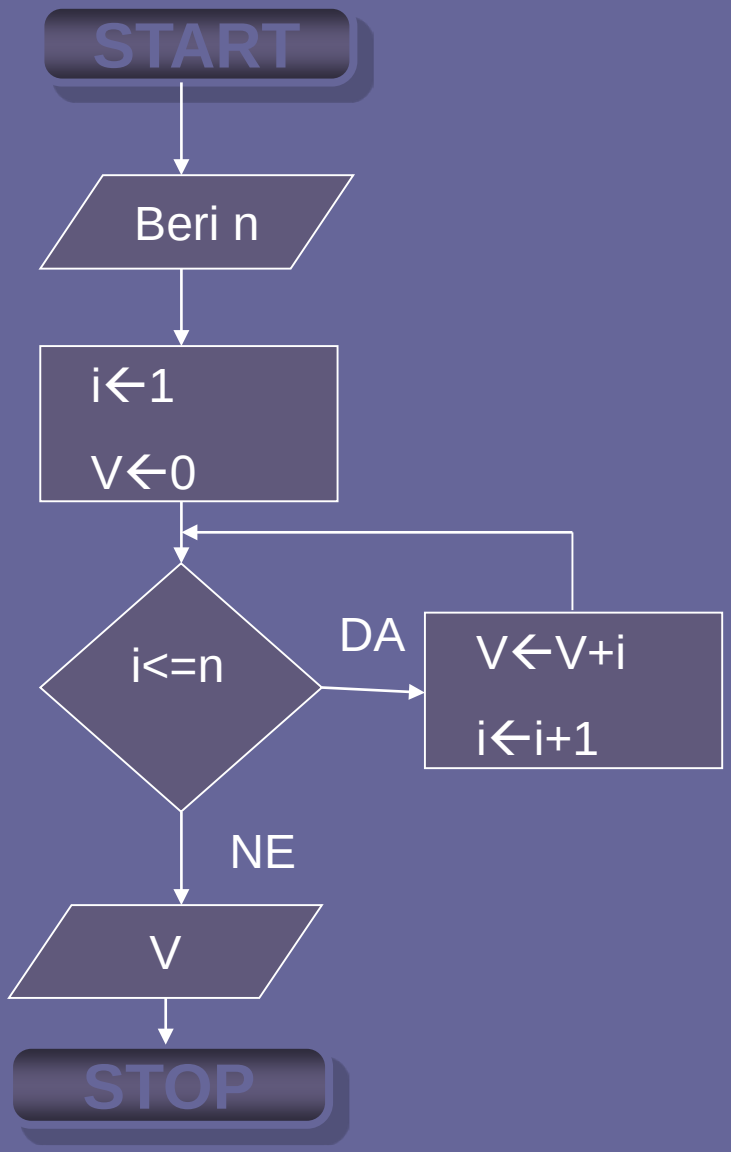


Diagram poteka za izračun vsote prvih n-števil



TABELARIČNE SPREMENLJIVKE

- Kadar imamo veliko število spremenljivk, vsebujejo nekatere spremenljivke podatke enakih podatkovnih tipov in z njimi v algoritmu ravnamo podobno. V algoritmu ni nobene razlike med postopkom, ki ga izvajamo s prvo takšno spremenljivko, drugo, tretjo, …
- Ko imamo veliko število spremenljivk ali ne vemo, koliko bo podatkov v algoritmu uporabimo **TABELARIČNE SPREMENLJIVKE.**
- Tabelarične spremenljivke imajo skupno ime, med seboj pa jih ločujemo z indeksom (T1, T2, …,Tn)..

PRIMER:

i	Ti
1	24
2	9
3	7
4	3
5	68
6	69

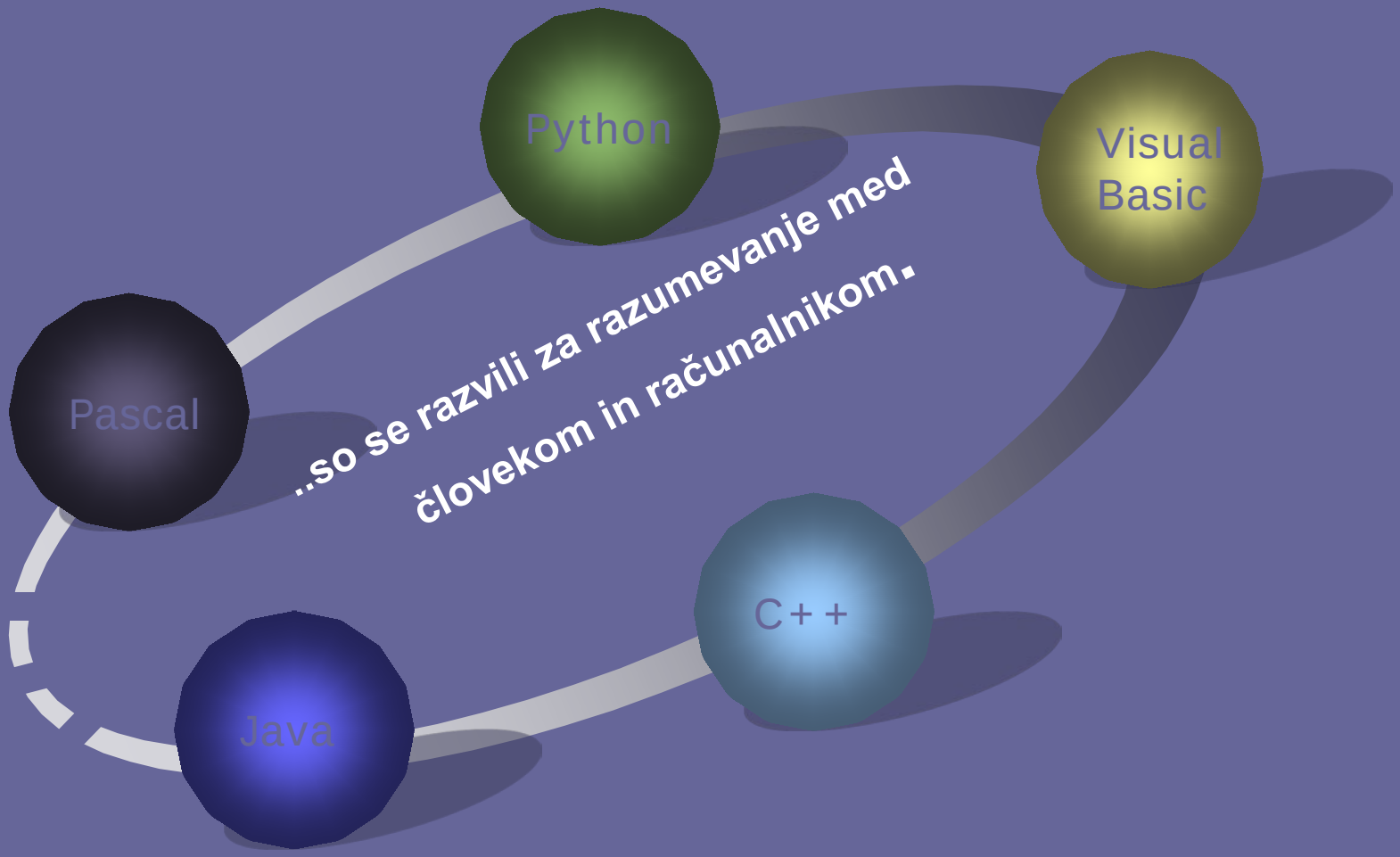
i = indeks (pove, katero je to število po vrsti)
T(i)=število

V algoritmih izvedemo postopek nad enim elementom, nato pa le spreminjamo indeks.

JEZIKI

- Ljudje komuniciramo med seboj v človeškem naravnem jeziku: slovenščini, ruščini, angleščini,…
- Vendar ti jeziki niso vedno uporabni. Za posebne namene so ljudje ustvarili posebne jezike: esperanto, jezik kemijskih formul, jezik aritmetičnih izrazov,… To so t.i. umetni jeziki.
- Ker tudi za komunikacijo z računalnikom naravni jezik ni primeren, so nastali posebni programski jeziki.

PROGRAMSKI JEZIKI



Programski jezik..

mora omogočati

Opis PROBLEMA

Katere podatke potrebujemo

Kje dobimo podatke

Kam shranimo podatke

Koliko je podatkov

Kakšne operacije bomo opravljali z njimi

Kako bomo izrazili rezultat

Opis POSTOPKA za rešitev problema

vsebuje opis korakov, ki nas od izhodiščnih podatkov pripeljejo k rezultatom.

Glede na opis postopka delimo programske jezike na:

- Vsebujejo izrazna sredstva za opredelitev podatkov in algoritmičnih gradnikov za opis postopka rešitve.
- Zahtevajo opis rešitve.
- Ta oblika programiranja je pogostejša in več je tudi postopkovno usmerjenih jezikov.
- Predstavniki so: Java, Python, Pascal, Basic…

- Vsebujejo sredstva za opis podatkov in relacije med njimi.
- Postopek reševanja opravi program, ki poišče ustrezen postopek, ki prevede podatke v rezultat.
- Bliže človeku, ker zahtevajo le opis problema, ni pa se nam treba ukvarjati s postopkom reševanja.
- Predstavnik je programski jezik Prolog

Programski jeziki in računalnik

- Ukazi, ki jih sprejema računalnik morajo biti predstavljeni z binarnimi znaki.
- Vsak ukaz mora biti zapisan tako, da ga računalnik prepozna in ve, kaj mora narediti, da ga izvrši.

Strojni jezik	Zbirni jezik ali assembler
Ukazi so zapisani v težko razumljivem dvojiškem zapisu.	Je že pomik k ljudem razumljivejšim jezikom.
	Ukaze opisujemo s simboli (LA-prenos, A-seštevanje, …). Namesto dvojiških naslovov v pomnilniku, se tu uporabljajo imena spremenljivk.
Takšen program je odvisen od vrste mikroprocesorja.	Težka tvorba in vzdrževanje velikih programov

PRIMER

Prenos vrednosti neke spremenljivke iz pomnilnika v ALE za neki računalnik

1001 1110 **LA C1**

Ukaz v strojnem jeziku, za prenos vrednosti iz pomnilnika v aritmetično logično enoto.

Naslov lokacije v pomnilniku, iz katere naj se vrednost prenese v aritmetično logično enoto.

Višji programski jeziki	Jeziki 4. generacije	Jeziki umetne inteligence
Da bi se programiranje še bolj približalo človeku, več ukazov zbirnega jezika združimo v en ukaz.	Rač. uporabljamo danes za reševanje ozko v določeno področje usmerjenih problemov, kot so upravljanje baz, upravljanje likovnih in zvočnih predstavitev,…	So nastali kot plod prizadevanj, da bi prog. jezike čim bolj približali človeku.
Tako so nastali višji programski jeziki, kot so Java, Python, Pascal, Fortran, Lisp, Algol…	Tako so nastali posebni jeziki in med najbolj znane sodi poizvedovalni jezik SQL.	Podobni so naravnim jezikom in se uporabljajo pri tehnologijah umetne inteligence.
So neodvisni od mikroprocesorja.		Zasnovani so na angleškem jeziku, ker se le ta v rač. največ uporablja.
Programiranje z njimi je hitrejše in lažje.		

Prevajanje programskih jezikov

- **Vsak program napisan v višjem jeziku je treba prevesti v strojni jezik torej jezik ki ga računalnik edinega razume. Za to potrebujemo PREVAJALNI PROGRAM.**
- **Čim manj je programski jezik podoben strojnemu tem bolj so prevajalniki zapleteni.**

  zbirni jezik ➔ strojni jezik …… preprost prevajalnik

  jeziki UI ➔ strojni jezik …… zapleten prevajalnik

- **Vendar prevod dobimo le, če je program SINTAKTIČNO PRAVILEN (- napisan v skladu s pravili tega jezika). Prevajalnik nas na te vrste napake opozori.**

- **Če prevajalnik program prevede, vendar delujoč preveden program ne daje pravih rezultatov, tedaj program vsebuje pomensko ali SEMANTIČNO NAPAKO. Prevajalnik te vrste napak ne zna odkriti.**

Prevajalniki in tolmači

Poznamo ti dve vrsti prevajalnih programov, ki prevajajo programe napisane v višjem programskem jeziku.

PREVAJALNIK	TOLMAČ (interpreter)
Naenkrat prevede celoten program v strojni jezik. Preveden program lahko večkrat uporabimo.	Ukaze sproti tolmači v ukaze strojnega jezika.
Pri izvajanju uporabimo le prevedeni program (izvorne kode ne potrebujemo več).	Ker se prevod ne shranjuje, potrebujemo pri naslednjem zagonu programa zopet originalni program napisan v višjem programskem jeziku.
Hitrejše izvajanje programa.	Lažje odkrivanje napak v programih.

Kako razviti algoritem s čim manj napakami?

STRUKTURIRANO

Začnemo z grobo, enostavno zamislijo algoritma, ki jo nato postopoma razčlenjujemo.

Problem razgrajujemo na podprobleme, dokler ne pridemo do problemov, ki jih lahko rešimo preprosto in brez napak.

DOGODKOVNO

Določen program se sproži, ko se izvrši nek dogodek.

Značilno za OS s slikovnim uporabniškim vmesnikom.

OBJEKTNO

Osnovni element je RAZRED. OBJEKTI so zgrajeni iz podatkov in metod.

Predmeti so organizirani hierarhično (od bolj splošnih k podrobnejšim).

Objekti dedujejo podatke in metode od svojih predhodnikov.

Učinkovito programiranje (vnaprej pripravljene razrede predmetov uporabimo v novih programih).

PYTHON

- Python je brezplačen in enostaven programski jezik.
- Je interaktiven, objektno usmerjen skriptni programski jezik in sodi med odprto programsko opremo.
- Njegovi ukazi se med izvajanjem tolmačijo v strojni jezik.
- Razvit je bil v devetdesetih letih prejšnjega stoletja na Nizozemskem.