

OSNOVE RAČUNALNIŠTVA IN INFORMATIKE



doc. dr. ANDREJ TARANENKO

Matematika
Izobraževalno računalništvo + ...

Osnovna literatura



- Viljem Žumer, Janez Brest: *Strukturirano in objektno usmerjeno programiranje v C++ (1. del)*, FERI UM.
- Edward Scheinerman: *C++ for mathematicians*, Chapman & Hall
- Derek M. Capper: *Introducing C++ for scientists, engineers and mathematicians*, Springer

<http://matematika-racunalnistvo.fnm.uni-mb.si/>

3

- **ŠTUDIJ**
 - Moji predmeti
 - ✦ Študijski program: Matematika (Izobraževalno računalništvo)
 - ✦ Osnove računalništva in informatike

- **ČLANI ODDELKA**
 - prof. dr. Aleksander Vesel
 - doc. dr. Andrej Taranenko
 - asist. dr. Igor Pesek

- **ZUNANJI SODELAVCI**
 - Eva Ferik

Vsebina tega dela predavanj

4

programski jezik C++

1. Uvod v programiranje

5

KLJUČNI OSNOVNI POJMI, KI SE
POJAVLJAJO V NADALJEVANJU

Kaj je računalništvo?

6

Kaj je **algoritem**?

- Nedvoumen postopek za rešitev problema v končno mnogo korakih.
- Je metoda, kako korak za korakom rešimo nalogo.

Računalništvo je veda o **algoritmih**, vključno z

- njihovimi formalnimi in matematičnimi lastnostmi
- njihovo realizacijo s strojno opremo
- njihovo realizacijo z jeziki
- njihovo uporabo

Kaj je programiranje?

7

- je zapis algoritma v nekem (programskem) jeziku
- je sporočanje algoritma računalniku

PRIMER

1. preberi vrednost x
2. preberi vrednost y
3. izračunaj vsoto x in y ter so shrani v z
4. izpiši vsoto na ekran

1. cin >> x;
2. cin >> y;
3. z = x + y;
4. cout << "Vsota je " << z;

Kaj je programski jezik?

8

- je umetni jezik, ki je razvit z namenom:
 - da ga razume neka naprava,
 - da izraža izračune, ki jih lahko izvede naprava,
 - je skupek sintaktičnih in semantičnih pravil
- sintaksa (oz. skladnja) jezika
 - pravila, ki določajo, kako tvorimo stavke v tem jeziku
- semantika jezika
 - določa pomen posameznih stavkov in njegovih elementov

Osnovni koraki programiranja

9

Potek programiranja:

1. Določi nalogo.
2. Najdi algoritem, ki nalogo reši.
3. Sprogramiraj algoritem v programskem jeziku.
4. Preveri in testiraj dobljeni program.

Primer

10

- Zapiši algoritem, ki v danem besedilu prešteje, kolikokrat se pojavi črka 'a'.

Nekaj uvodnih pojmov (1)

11

- računalnik
 - procesor
 - pomnilnik
 - vhodno/izhodne naprave

- operacijski sistem
 - nadzira in upravlja vire računalnika
 - nudi uporabnikom orodja za delo
 - × urejevalnik besedil
 - × prevajalnik

Nekaj uvodnih pojmov (2)

12

- izvorna koda
- izvorna datoteka
- prevajalnik
- izvedljiva koda
- izvedljiva datoteka

Prevajalnik

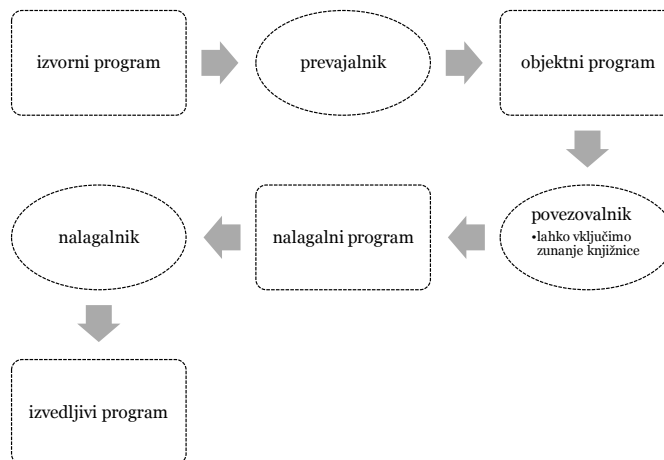
13

- je program, ki pretvori izvorni program (običajno zapisan v nekem VPJ) v drug (običajno strojni) jezik
- rezultat je izvedljivi program
 - računalnik ga razume
 - računalnik ga zna izvesti
- prevedemo celoten program in ga nato izvedemo



Kompleksnejša shema prevajanja

14



Kaj zahteva programiranje?

15

- **programiranje je ustvarjalen proces**
 - ni točnega navodila, kako programirati
- **običajno poteka v dveh fazah**
 - 1. faza (težji del) – reševanje problema
 - ✦ rezultat te faze je algoritem, ki reši dani problem
 - 2. faza – implementacija algoritma
 - ✦ algoritem pridobljen v prejšnji fazi zapišemo v izbranem programskem jeziku

Faza reševanja problema

16

- **natančno preberi problem**
 - popolno razumevanje problema je ključno pri iskanju pravilne rešitve
- **preveri, ali je problem natančno definiran**
 - Kateri podatki so podani s problemom?
 - Kaj je želena rešitev problema oz. cilj?
 - V kakšni obliki moram podati rešitev?
- **ne preskoči te faze – ne prični takoj z implementacijo**
 - prihranimo na času
 - se izognemo nepotrebni napakam
 - preveri, ali algoritem deluje pravilno

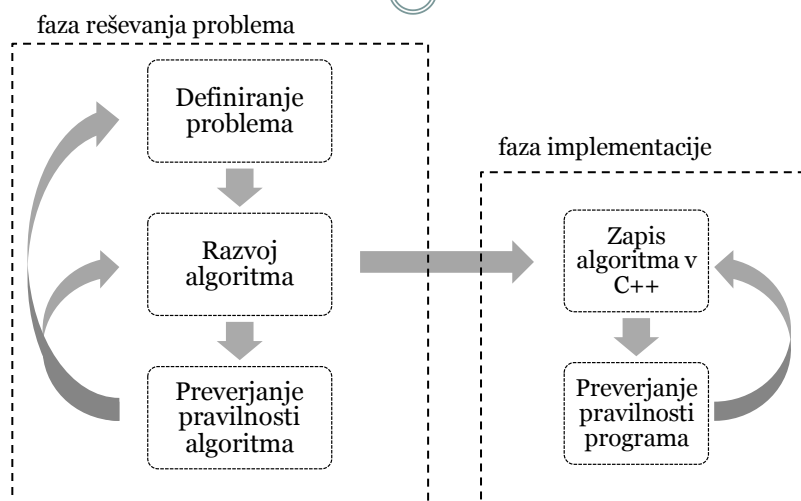
Faza implementacije

17

- algoritem dobljen v predhodni fazi pretvori v programski jezik
 - vaja dela mojstra – ta proces postaja s časom vedno lažji
- izvorni program (kodo) prevede s prevajalnikom
 - to bo preverilo tudi, ali se držimo sintaktičnih pravil jezika
- zaženi izvedljivi program in ga testiraj z znanimi podatki
 - preveri pravilnost rezultatov
- odpravi morebitne napake

Programiranje

18



Integrirano razvojno okolje

19

- **program, ki omogoča programiranje**
 - podpira vse korake v procesu programiranja
 - mi bomo uporabljali Code::Blocks
- **nudi vse možnosti razvoja programske opreme**
 - urejevalnik
 - prevajalnik
 - povezovalnik
 - nalagalnik
 - razhroščevalnik
 - ...

