

Programiranje 2

Uvodno predavanje

dr Đorđe Obradović

Singidunum

Uvod

Programiranje

- Šta je programiranje?

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskell
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskell
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskell
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskell
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

Programski jezici

- Python
- Java
- Javascript
- Fortran
- Assembler
- Cobol
- Prolog
- Matlab
- C
- C++
- C#
- GO
- ruby
- Haskel
- Lisp

C++

- Zašto da uopšte programiramo u C++?

Python

```
0  a = [1, 2, 3, 13, 6, 9, 3, 8, 5, 6]
1
2  def sum(niz):
3      ret = 0
4      for el in niz:
5          ret += el
6      return ret
7
8  print sum(a)
```

56

Java

```
0  class test {  
1  
2      public static int sum(int[] niz){  
3          int ret = 0;  
4          for (int e : niz)  
5              ret += e;  
6          return ret;  
7      }  
8  
9      public static void main(String[] args) {  
10         int[] a = {1, 2, 3, 13, 6, 9, 3, 8, 5, 6};  
11         System.out.println(sum(a));  
12     }  
13 }
```

Javascript

```
0  function sum(niz){  
1      var ret = 0;  
2      for (var e in niz)  
3          ret += niz[e];  
4      return ret;  
5  }  
6  var a = [1, 2, 3, 13, 6, 9, 3, 8, 5, 6];  
7  console.log(sum(a));
```

Interesantan primer - ulaz



Interesantan primer - implementacija python

```
0
1  import cv2
2
3  imagePath = 'srbija.jpg'
4  cascPath = 'haarcascade_frontalface_default.xml'
5
6  faceCascade = cv2.CascadeClassifier(cascPath)
7  image = cv2.imread(imagePath)
8  gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
9
10 faces = faceCascade.detectMultiScale(
11     gray,
12     scaleFactor=1.1,
13     minNeighbors=5,
14     minSize=(30, 30),
15     flags = cv2.cv.CV_HAAR_SCALE_IMAGE
16 )
17
18 print "Found {0} faces!".format(len(faces))
19
20 for (x, y, w, h) in faces:
21     cv2.rectangle(image, (x, y), (x+w, y+h), (0, 255, 0), 2)
22
23 cv2.imwrite("faces-srbija.jpg", image)
24 cv2.imshow("Faces found" ,image)
25 cv2.waitKey(0)
```


Interesantan primer - rezultat



OpenCV



OpenCV biblioteka za python, java, C#, matlab...

Izvršavnje na platformama: Windows, Linux, Mac OS, iOS i Android.

Dizajn i implementacija: računarska efikasnost i primena u realnom vremenu

Napisana: u optimizovanom C/C++

Nginx



Web server: - prva verzija 2004. -
reverzni proxy, - load balanser i - http
keš

Operativni sistemi: - Windows - Linux -
Mac OS

Napisan u programskom jeziku C zbog
efikasnosti.

MongoDB



NoSQL baza: - prva verzija 2009. -
poslednja verzija septembar 2017

Napisan u programskim jezicima:
C++, C i Javascript.

Osnovne karakteristike jezika C++

Karakteristitke

- programski jezik opšte namene
- imperativno/proceduralno programiranje (iskazi pomoću kojih se menja stanje)
- objektno orijentisani pristup
- generičko programiranje (korišćenje šablona)
- upravljanje memorijom na najnižem nivou
- najčešće se koristi za:
 - sistemsko programiranja
 - za sisteme koji treba da rade u uslovima strogih vremenskih i prostornih ograničenja
 - implementacijama sistema za upravljanje podacima
 - u aplikacijama kojih su performanse veoma bitne

Evolucija jezika C++

- [1979.] Bjarne Stroustrup - C sa klasama
 - Simula programski jezik je bio pogodan za velike sisteme
 - C je izabran zbog opšte namene i brzine a ideje o klasama su preuzete iz Simula koji je bio suviše spor
- [1983.] 'C with classes' je preimenovan u C++, proširenje sa:
 - virtuelnim funkcijama,
 - preklapanjem funkcija,
 - referencama i konstantama
- [1985.] prvo izdanje knjige The C++ Programming Language, prva referenca jezika
- [1989.] C++ 2.0 a drugo izdanje knjige koja opisuje verziju 2.0 izašlo je 1991.
 - Višestruko nasleđivanje, abstraktne klase,
 - statički atributi
- [2011.] C++11 standard u kojem je pokušana standardizacija jezika sa mnogim proširenjima koji uključuju standardnu biblioteku
- [2017.] C++ je treći najpopularniji programski jezik iza Jave i C-a

Organizacija programa

- programi se pišu kao tekstualne datoteke.
- datoteke sa ekstenzijom .h sadrže opise zaglavlja
- datoteke sa ekstezijom .cpp sadrže definicije i implementacije: klasa, globalnih funkcija i globalnih varijabli
- sva zaglavlja se deklarišu na početku .cpp datoteka

Korišćenje

- .cpp datoteke se kompajlerima prevode u
- .obj datoteke koje sadrže mašinski kod (naredbe koje su prilagođene mašinama)
- više .obj datoteka se povezuju u jedan program pomoću linkera (program koji razrešava sve veze između elemenata u .obj datotekama)
- rezultat je jedna izvršna datoteka (na windows OS obično imaju ekstenziju .exe)

Zaglavlja

Tekstualne datoteke koje opisuju elemente za povezivanje više različitih modula ili .cpp fajlovima

```
0
1  #include <iostream>
2
3  int main()
4  {
5      std::cout << "Hello, world!\\n";
6      return 0;
7  }
```

Biblioteke

Veoma bitna karakteristika jezika je mogućnost ponovnog korišćenja implementiranih funkcija i klasa.

Primer je biblioteka matematičkih funkcija. Trigonometrijske funkcije:

- cos
- sin
- tan
- acos
- asin
- atan
- atan2

Struktura C++ programa:

```
0  // my first program in C++
1  #include <iostream>
2
3  int main()
4  {
5      std::cout << "Hello World!";
6  }
```

Tutorial - OSNOVE

Komentari

```
0  /* moj  
1    prvi  
2    program u C++  
3  */  
4  #include <iostream>  
5  
6  int main(){  
7    std::cout << "Hello World!";  
8  }
```

Namespace

```
0 // moj drugi program u C++
1 #include <iostream>
2
3 int main ()
4 {
5     std::cout << "Hello World! ";
6     std::cout << "I'm a C++ program";
7 }
```

Može ovako:

```
0 // moj drugi program u C++
1 #include <iostream>
2 using namespace std;
3
4 int main ()
5 {
6     cout << "Hello World! ";
7     cout << "I'm a C++ program";
8 }
```

Ključne reči

najčešće rezervisane reči:

```
0 and, auto, bool, break, case, catch, char, class, const,  
1 continue, decltype, default, do, double, else, enum, false,  
2 float, for, if, int, long, namespace, new, not, nullptr, or,  
3 private, protected, public, return, short, signed, sizeof,  
4 static, struct, switch, template, this, throw, true, try,  
5 typedef, void, while
```


Varijable

Varijable nazivi za promenljive.

C++ je tipiziran programski jezik.

Sve varijable u trenutku prevođenja moraju da budu tipizirane.

Tipovi podataka

Osnovni/ugrađeni tipovi podataka:

- char
- int
- float
- double
- long double
- bool
- void
- nullptr

Deklaracije varijabli

```
0  int a;  
1  int b;  
2  bool c;  
3  int d, e, f;
```

Jednostavan program

```
0  #include <iostream>
1  using namespace std;
2
3  int main ()
4  {
5      // deklaracija varijabli:
6      int a, b;
7      int result;
8
9      // obrada:
10     a = 5;
11     b = 2;
12     a = a + 1;
13     result = a - b;
14
15     // prikaz rezultata:
16     cout << result;
17
18     // zavrsetak:
19     return 0;
20 }
```

Inicijalizacija varijabli

```
0  int main ()
1  {
2      int x = 0;
3      int y{2};
4      int rez = x+y;
5      cout << rez;
6  }
```

Stringovi

```
0  #include <iostream>
1  #include <string>
2  using namespace std;
3
4  int main ()
5  {
6      string mystring;
7      mystring = "This is a string";
8      cout << mystring;
9      return 0;
10 }
```

Konstante

```
0  75          // decimal
1  0113        // octal
2  0x4b        // hexadecimal
```

ili

```
0  75          // int
1  75u         // unsigned int
2  75l         // long
3  75ul        // unsigned long
4  75lu        // unsigned long
```

karakter i string literali

```
0  'z'
1  'p'
2  "Hello world"
3  "How do you do?"
```

Posebni 'escape' karakteri

```
0  \n  newline
1  \r  carriage return
2  \t  tab
```


Imenovane konstante

```
0  const double pi = 3.1415926;  
1  const char tab = '\\t';
```

Primer korišćenja

```
0  #include <iostream>
1  using namespace std;
2
3  const double pi = 3.14159;
4  const char newline = '\\n';
5
6  int main ()
7  {
8      double r=5.0;           // radius
9      double circle;
10
11     circle = 2 * pi * r;
12     cout << circle;
13     cout << newline;
14 }
```

Preprocessor definicije (#define)

```
0  #include <iostream>
1  using namespace std;
2
3  #define PI 3.14159
4  #define NEWLINE '\\n'
5
6  int main ()
7  {
8      double r=5.0;           // radius
9      double circle;
10
11     circle = 2 * PI * r;
12     cout << circle;
13     cout << NEWLINE;
14 }
```

Operatori

Dodela vrednosti

```
0 x = 5;
```

```
1
```

```
2 x = y;
```

Primer

```
0  #include <iostream>
1  using namespace std;
2
3  int main ()
4  {
5      int a, b;           // a:?, b:?
6      a = 10;             // a:10, b:?
7      b = 4;              // a:10, b:4
8      a = b;              // a:4, b:4
9      b = 7;              // a:4, b:7
10
11     cout << "a:";
12     cout << a;
```

Aritmetički operatori (+, -, *, /, %)

5 osnovnih aritmetički operatora:

+, -, *, /, %

Složeni operatori dodele vrednosti

Nekada se zbog jednostavnijeg/efikasnijeg kompajliranja koriste i operatori:

$+=$, $-=$, $*=$, $/=$, $\%=$

```
0  #include <iostream>
1  using namespace std;
2
3  int main ()
4  {
5      int a, b=3;
6      a = b;
7      a+=2;           // equivalent to a=a+2
8      cout << a;
9  }
```

Increment and decrement ($++$, $-$)

Da ih ne preskočimo iako verovatno znate šta su!

Relacioni operatori

`==` JEDNAKO

`!=` NIJE JEDNAKO

`>`, `<`, `>=`, `<=`

Logički operatori

! NE

&& I

|| ILI

Uslovni operatori

```
0  7==5 ? 4 : 3      // evaluates to 3, since 7 is not equal to 5.
1  7==5+2 ? 4 : 3    // evaluates to 4, since 7 is equal to 5+2.
2  5>3 ? a : b       // evaluates to the value of a, since 5 is greater t
3  a>b ? a : b
```

Bit operatori ($\&$, $|$, \wedge , \sim , \ll , \gg)

($\&$, $|$, \wedge , \sim , \ll , \gg)

Cast operator

```
0  int i;  
1  float f = 3.14;  
2  i = (int) f;
```

Iskazi i kontrola toka programa IF

```
0  if (x == 100)
1  {
2      cout << "x is ";
3      cout << x;
4  }
```

IF ELSE

```
0  if (x == 100)
1      cout << "x is 100";
2  else
3      cout << "x is not 100";

0  if (x > 0)
1      cout << "x is positive";
2  else if (x < 0)
3      cout << "x is negative";
4  else
5      cout << "x is 0";
```

WHILE

```
0  #include <iostream>
1  using namespace std;
2
3  int main ()
4  {
5      int n = 10;
6
7      while (n>0) {
8          cout << n << ", ";
9          --n;
10     }
11
12     cout << "liftoff!\n";
```


DO-WHILE

```
0  #include <iostream>
1  #include <string>
2  using namespace std;
3
4  int main ()
5  {
6      string str;
7      do {
8          cout << "Enter text: ";
9          getline (cin,str);
10         cout << "You entered: " << str << '\n';
11     } while (str != "goodbye");
12 }
```

FOR

```
0  #include <iostream>
1  using namespace std;
2
3  int main ()
4  {
5      for (int n=10; n>0; n--) {
6          cout << n << ", ";
7      }
8      cout << "\n";
9  }
```

FOR CONTINUE

```
0  #include <iostream>
1  using namespace std;
2
3  int main ()
4  {
5      for (int n=10; n>0; n--) {
6          if (n==5) continue;
7          cout << n << ", ";
8      }
9      cout << "\n";
10 }
```

SWITCH

```
0  switch (expression)
1  {
2      case constant1:
3          group-of-statements-1;
4          break;
5      case constant2:
6          group-of-statements-2;
7          break;
8      default:
9          default-group-of-statements
10 }
```

FUNKCIJE

```
0  #include <iostream>
1  using namespace std;
2
3  int zbir(int a, int b)
4  {
5      int r;
6      r=a+b;
7      return r;
8  }
9
10 int main ()
11 {
12     int z;
```