



www.puskice.org

Programski jezici

Primeri teoretskih pitanja

<!-- Ispravke, sugestije, mišljenja i ostalo šalјite na download@puskice.org -->

2013.

1. Razlika između imperativnih i funkcionalnih jezika:

- Modeli na kojima se zasnivaju
- Program (imperativni: niz instrukcija, funkcionalni: funkcija)
- Ključna operacija (imperativni: dodeljivanje, funkcionalni: primena funkcije)
- Dodela (imperativni: eksplicitno, funkcionalni: nema eksplicitne dodele)
- Rad sa memorijom (imperativni: eksplicitni, funkcionalni: implicitni)
- Tipovi (imperativni: tipiziran jezik, funkcionalni: bez tipova (samo dva osnovna: atom i lista))

1. Imperativna paradigam podržava:

- Bazne
- Proceduralne
- Strukturne apstrakcije

2. Ključna operacija imperativne paradigam:

Dodeljivanje

3. Ključna operacija objektno-orijentisane paradigam je:

Prenošenje poruke

4. Koji jezici apstrauju Von Neumanna

Imperativni

5. Programski jezik LISP je:

Funkcionalni jezik

6. Programski jezik Prolog je:

Logički jezik

7. Za leksičku specifikaciju programa koriste se:

Regularni izrazi

8. Za formalan opis sintakse programskih jezika se koriste:

Konteksno-slobodne gramatike (CFG)

9. Koji od navedenih pristupa se koristi za formalin opis semantike p.j.-a?

- Aksiomatski
- Denotacioni
- Translacioni
- Operacioni

10. Da li interpreter prevodi program iz izvornog u izvršni kod?

Ne

11. Programske paradigme obezbeđuju model projektovanja/implementacije softverskih sistema

Da

12. Baze apstrakcije su:

- Promenljive
- Elementarni tipovi podataka
- Operacija dodeljivanja

13. Šta su alijasi?

Dve ili više promenljivih koje pristupaju jednoj memorisnoj lokaciji

14. Pomoću čega se alijasi kreiraju?

Pointera i referentnih promenljivih

15. Koji su referentni tipovi podataka?

Klasa, interfejs, delegate, niz, predefinisani objekat i string

16. Koji su vrednosni tipovi podataka?

Struct, enum, int, short, long, float, double, decimal, byte, char, bool

17. Koji su to korisnički definisani tipovi?

Enum, struct, niz, klasa

18. Statička promenljiva alocira se u statičkom segmentu memorije:

Samo jednom

19. Životni vek statičke promenljive je:

Vreme izvršenja programa

20. Vrednost statičke promenljive je:

Perzistentna

21. Statički doseg promenljive definiše se u terminima

Leksičke strukture programa

22. Doseg promenljive je:

Segment koda programa u kome je promenljiva poznata i može da se koristi

23. Dinamički doseg promenljive zasniva se na:

Sekvenci poziva potprograma

24. Problem tzv. "garbage" dinamičke promenljive dele p.j. u kojima se dinamička promenljiva delocira:

Eksplisitno

25. Problem “Lost” heap-dinamičke promenljive
Alocirana promenljiva kojoj se ne može pristupiti
26. Koje od sledećih situacija kreira “dangling pointer”?
Posle delokacije dinamičke promenljive, postoji pointer koji sadrži adresu heap-dinamičke promenljive.
27. Predefinisani tipovi podataka dati su u:
CTS (Common Type System)
28. Životni vek promenljive:
Vreme koje ona provede vezana za neku memorijsku lokaciju
29. Vrednost eksplicitne heap-dinamičke promenljive je:
Perzistentna
30. Vrednost implicitne heap-dinamičke promenljive :
sve svaki put realizuje
31. Interpretacija realnog tipa (double) u C# je:
 - a) binarna
 - b) preko mantise i promenljive
 - c) BCD codePreko mantise i promenljive
32. Interpretacija decimalnog i znakvonog tipa u C# je:
BCD Code
33. Delokacija Garbage promenljive u C# je:
Eksplicitna
34. Delokacija promenljivih u C# je:
Implicitna
35. BCD je:
4-bitni kod
36. Vrednost stack-dinamičke promenljive:
Reinicijalizuje se pri svakom pozivu funkcije
37. Vrednost stack-dinamičke promenljive:
Nije perzistentna
38. Šta je .NET Framework?
Softver koji povezuje informacije, ljude, sisteme i uređaje.
39. Najvažniji deo .NET platform je:
Framework
40. .NET je:
I razvojno i izvršno okurženje
41. Koji deo .NET Frameworka sadrži predefinisane tipove podataka?
CTS – Common Type System
42. Da li je .NET operativni system
Ne
43. Koncepti objektno-orijentisanog programiranja:

- Nasleđivanje, Učarenje, Polimorfizam
44. Koja je bazna klasa u C#?
System.Object
 45. Da li u C# nasleđivanje može da bude višestruko?
Ne može
 46. Konstruktori se ne nasleđuju, izvršavaju se hijerarijski i mogu imati modifikator pristupa
 47. Izvedena klasa može da nasledi sve osim:
Konstruktora i destruktora
 48. Mehanizam koji obezbeđuje vidljivost interfejsa ATP u C++:
Public klauzula
 49. Imenovane inkapsulacije:
C++ namespace, C# namespace, Java packages, ADA packages
 50. Inkapsulaciona konstrukcija C# je:
Assembly bi trebalo da bude odgovor, a samo inkapsulacija bi bila klasa ????
 51. Inkapsulaciona konstrukcija C++ je:
U Header file-u
 52. Imenovana inkapsulacija:
Namespace
 53. Konstrukcionu inkapsulaciju:
Assembly
 54. Mehanizam za inkapsulaciju implementacije apstraktonog tipa podataka u C# je:
klasa
 55. Šta se koristi za "Interfejs entiteta" u C++?
Public klauzula
 56. Šta se koristi za skrivanje entiteta u C++?
Private
 57. Šta se koristi za nasleđivanje u C++?
Protected
 58. Šta znači internal?
Klase istog asemblija
 59. DTD je skraćenica za:
Document Type Definition
 60. DTD:
Nasleđen od SGML-a
Poseban jezik za opis structure
Vrlo ograničene mogućnosti za definisanje tipova
 61. XSLT je skraćenica za:
eXtensible Stylesheet Language Transformation
 62. XSKT:
Na ulazu ima XML document i XSLT stylesheet (schema, XSD), a na izlazu XML, HTML, tekst
 63. Kardinalnost elemenata u XML šemi specificira se preko:

minOccurs i maxOccurs

64. SGML je:

Standard za definisanje i reprezentovanje structure različitih tipova ečeltronskih dokumenata

65. XSLT je:

- a) deklarativan jezik
- b) proceduralan jezik
- c) logički jezik

Deklarativni jezi

66. Konstruktor all u XML šemi definiše:

Brupu u kojoj svi elemnti mogu maksimalno jednom da se pojave

67. XSLT Procesor za transformaciju elementa koristi

- a) XML šemu
- b) HTML dokument
- c) XSLT stylesheet dokument

odgovor: a) i c)

68. Sequence:

Uređena grupa elemenata, svaki element obavezan i jednoznačan

69. Choice

Mogu se izvlačiti pojedinačni elementi

70. XSLT processor za transformaciju elemenata koristi:

XML šemu

XSLT stylesheet document

71. Atribut match u templejtu element `<xsl:template match=""/>`

Povezuje templejt sa korenim elementom

72. XML Reader reprezentuje:

Pull model XML Parser

73. Za parsiranje XML dokumenta mogu se koristiti:

Sax i dom

74. Da li je sax w3c standard?

Ne

75. Kada se koristi DOM parser?

Kada se zahteva obrada većine elemenata, kada se zahteva dinamički pristup i manipulacija

76. Kada se koristi SAX parser?

Kada ne zahteva dinamički pristup i kod obrade prostih dokumenata sa puno podataka

77. HTML se koristi za:

Formatiranje i prikazivanje dokumenata

78. XML se koristi za:

Struktuiranje dokumenata

79. Tip promenljive je apstraktan ako su klijentima vidljive samo:

Skup operacija i naziv tipa

80. Koja su tvrđenja tačna?
Svaki XML element mora da ima početni i krajnji tag
XML dokument ima jedan i samo jedan koreni element
81. XSLT je XML dokument?
Da
82. Sax je:
Push model
83. XML šema specificira:
Skup predefinisanih pravila za opis structure XML dokumenta
84. XML je podskup:
SGML-a
85. Da bi XML document bio validan on mora biti:
Konzistentan sa strukturom definisanom u opisu tipa dokumenta
Dobro oformljen
86. XML je skraćenica od:
eXtensible Markup Language
87. Koji se standardi koriste za definisanje tipova podataka XML dokumenta?
DTD i XSD
88. U slučaju obrade XML dokumenta sa prostom strukturom koja sadrži veliki obim podataka i ako se ne zahteva dinamički pristup dokumentima, pogodnije je koristiti:
SAX parser
89. Sax model parsira dokumentil formira memorijsko stablo čvorova koje reprezentuje XML document
Ne
90. DTD je XML document
NE
91. Da bi XML document bio dobro oformljen on mora biti:
Sintaksno korektan
92. Koja od navedenih tvrđenja su tačna?
XML document je samoopisujući i platformski nezavisan
Dozvoljeno je višestruko ugnježdavanje XML dokumenta
93. Delokacija promenljivih u C# može biti:
Implicitna i eksplicitna
94. U C# su modifikatori pristupa, a u C++ su klauzule
95. C# kompajler vrši translaciju programa u:
a) mašinski jezik
b) bytecode
c) IL (Intermediate Language)
IL (Intermediate Language)
96. Da li je IL viši programski jezik?

ne

97. DOM parseri u .NETu su:

MSXML parser

XML document

98. DOM parser se koristi kada se zahteva obrada većine elemenata, kada se zahteva dinamički pristup i manipulacija

99. Transportovanje objekata u web servisima

UDDI

100. Osnovni delovi SOAP poruke:

Header, Body, Envelope

101. Da li se formulacija XML scheme izvodi koriscenjem predefinisanih tagova?

ne

102. Karakteristike apstraktnog tipa podataka su>

Učaurjenje

Skirvanje informacija

103. Strukture apstrakcije u programskim jezicima su:

Strukture podataka

Upravljačke sturkture

104. Koji od navedenih mehanizama se koristi za skrivanje informacija u C#?

Za svaku članicu definiše se vrsta pristupa

105. Koji od navedenih tipova podataka su apstraktni tipovi podataka?

106. W3C standardi za definisanje XML dokumenata su:

XML schema i DTD

107. Da li je XML dokument dobro oformljen:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<2006RezultatiTesta mesec="januar">
```

```
  <Polagalo>300</Polagalo>
```

```
  <Polozilo>199</Polozilo>
```

```
</2006RezultatiTesta>
```

Ne

108. Da li je XML dokument validan?

```
<?xml version="1.0" standalone="yes"?>
```

```
<PORUKE>
```

```
  <PORUKA datum="10.10.2005">
```

```
    <OD>Jovana</OD>
```

```
    <ZA>Nikola</ZA>
```

```
    <TEKST>Vidimo se sutra u 9</TEKST>
```

```
  </PORUKA>
```

```
</PORUKE>
```

```
</xml>
```

ne

109. Naziv protokola koji se koristi za prenos objekata preko web-a?

- a) WSDL
- b) DCOM**
- c) SOAP
- d) Ne znam

110. Koja su od sledećih tvrđenja tačna:

- a) XSLT koristi Xpath
- b) XSLT nije W3C standard
- c) XSLT je XML dokument

a)XSLT koristi Xpath

1. Najvažniji deo .net platform - odg : Framework
2. Koji deo .net Framework-a sadrži predefinisane tipove podataka- odg: Common Type System (CTS)
3. .net Framework je - odg : razvojno i izvrsno okruženje
4. Koji nisu korisnički definisani tipovi - ponudjeni odg : struktura, klasa, niz, string, delegat, enumeracija
5. Sta je leksički analizator(skener)- odg : prevodi tekst programa u niz tokena
6. Programski jezik LISP je - odg : Funkcionalan jezik
7. Karakteristike ATP - Inkapsulacija(ucaurenje) i skrivanje podataka
8. Inkapsulacija je - odg : odvajanje specifikacije ATP od njegove implementacije
9. Pass-by-reference - odg : INOUT ili kako je sad bilo ponudjeno IN OUT
10. Tehnologije web servisa - odg : XML i SOAP
11. Sta ulazi u XSLT procesor- odg XML
12. atribut match , pa da se zaokruži sta je tačno - odg : <xsl:template match="/">
13. konstruktor all u XML semi definise - odg : grupu u koju se svi elementi mogu pojaviti max jedanput
14. Protokol za prenos objekata preko weba - odg : SOAP
- Vrednost stack-dinamičke promenljive - odg : reinicijalizuje se pri svakom pozivu fje
15. Za leksičku specifikaciju PJ koristi se - odg : Regularni izrazi
16. Statika promenljiva alocira se u statickom segmentu memorije - odg : samo jednom
17. Tačna tvrdjenja za XML - odg: da ima krajnji tag i da mora da bude u skladu sa DTD
18. XPath - odg : koristi se za izdvajanje delova XML dokumenta na koje će biti primenjena transformacija
19. SAX parser - odg : veoma efikasan
20. Nisam sigurna da li je bilo sta mogu ili ne mogu da budu cvorovi-listovi , a ponudjeni odg su : dokument , elementi, text i comment
21. Imenovane inkapsulacije - odg : C namespace i C# namespace
22. da se zaokruže koji je dobar XML, i 2 su tačna odg ,pod a i pod b

23. SAX karakteristike - odg jedno je sigurno tacno da imaju sekvencijalan pristup, a jedno netacno tj da zahtevaju dinamički pristup, a ostalih se ne secam
24. bilo je nesto oko interpretera , i mislim da je tu tacan odg medju kod
25. ispravno napisam xml- odg: nesto sa velikim i malim slovima ne znam tacno

Prva Grupa:

1. Životni vek promenljive je:
 - a) segment koda programa u kome je promenljiva poznata i može se koristiti
 - b) vreme vezivanja promenljive za određenu memorijsku lokaciju, u toku izvršavanja programa
2. BCD je:
 - a) 8-bitni kod
 - b) 16-bitni kod
 - c) 4-bitni kod
3. Statički doseg promenljive zasniva se na
 - a) Sekvenci poziva potprograma
 - b) Leksičkoj strukturi programa
4. Vrednost stack dinamičke promenljive je
 - a) Perzistentna
 - b) Reinicijalizuje se pri svakom pozivu funkcije
 - c) Ne znam
5. Za leksičku specifikaciju programskih jezika koristi se
 - a) Binarno stablo
 - b) Konteksno-slobodna gramatika
 - c) Regularni izrazi
6. Mehanizam koji obezbeđuje vidljivost interfejsa ATP u C++ je:
 - a) public modifikator pristupa
 - b) public klauzula
 - c) protected kaluzula
7. Statička promenljiva alocira se u statičkom segmentu memorije
 - a) Više puta
 - b) Samo jednom
8. Koje su od navedenih imenovane inkapsulacije:
 - a) Java packages
 - b) C# assemblies
 - c) C# namespaces

Rešenja:

1. b)	2. c)	3. b)	4. b)	5. c)	6. b)	7. b)	8. a) i c)
-------	-------	-------	-------	-------	-------	-------	------------

Druga grupa:

- Doseg promenljive je:
 - segment koda programa u kome je promenljiva poznata i može se koristiti
 - vreme vezivanja promenljive za određenu memorijsku lokaciju, u toku izvršavanja programa
- Problem tzv. "garbage" dinamičke promenljive dele programski jezici u kojima se dinamička promenljiva dealocira:
 - Implicitno
 - EksPLICITNO
- Inkapsulaciona konstrukcija u C# je:
 - package
 - assembly
 - private klauzula
- Dinamički doseg promenljive zasniva se na
 - Sekvenci poziva potprograma
 - Leksičkoj strukturi programa
- Interna reprezentacija decimalnog tipa (Decimal type) u C# je:
 - Binarna
 - Preko mantise i eksponenta
 - BCD code
- Koja od sledećih situacija kreira "dangling pointer"
 - Dealociranje bloka koji je sadržao pointer na drugi postojeći memorijski blok
 - Posle dealokacije dinamičke promenljive postoji pointer koji sadrži adresu takve promenljive
 - Pointer koji pokazuje na dinamičku promenljivu, postavljen je na "null"
- Životni vek C static promenljive koja je deklarirana u nekoj funkciji jednak je
 - Vremenu izvršavanja programa
 - Vremenu izvršavanja funkcije
 - Ne može se predvideti
- Šta razlikuje funkcionalni od imperativnog programskog jezika
 - U funkcionalnom jeziku svi potprogrami se moraju deklarirati pomoću ključne reči function

- b) U funkcionalnom jeziku ne postoje promenljive i operacija dodeljivanja
- c) Ne postoji razlika
- d) U funkcionalnom jeziku ne postoji kontrolna struktura selekcije

Rešenja:

1. a)	2. b)	3. b)	4. a)	5. c)	6. b)	7. a)	8. b)
-------	-------	-------	-------	-------	-------	-------	-------

Drugi kolokvijum 2013

1. Razlika između imperativnih i funkcionalnih jezika:

- a) Nema razlike
- b) Kod funkcionalnih jezika svaki potprogram mora otpočeti rezervisanom rečju *function*
- c) Kod funkcionalnih jezika ne postoji naredba dodele

2. Koje su tehnologije web servisa:

- a) SOAP
- b) XML
- c) WSDL
- d) CORBA

3. Šta od ponuđenog ne može biti čvor list u DOM-u

- a) Komentar
- b) Tekst
- c) Dokument
- d) Element

4. Ispravno oformljen XML, ponuđena 4 komada.

5. Ne sećam se tačno pitanja, odgovor je bio a) XSLT je XML dokument

6. Koji je korektan način za referenciranje stylesheet dokumenta „pozdrav.xsl“

- a) `<stylesheet type = "text/xsl" href="pozdrav.xsl">`
- b) `<link type = "text/xsl" href="mystyle.xsl">`
- c) `? xml-stylesheet type = "text/xsl" href="pozdrav.xsl" ?>`

7. XDocument reprezentuje koji model XML parsera

- a) Pull model
- b) Push model
- c) DOM model

8. Koji su elementi xml fajla

- 1. Envelope
- 2. Header
- 3. nešto1
- 4. nešto2

Ispit 5.2.2005 - Grupa 1 i 2

p1. SAX model za parsiranje XML dokumenta je W3C standard?			A.
A. DA	B. NE		
2. DTD je skraćenica od?			B.
A. Direct Type Definition	B. Document Type Definition	C. Dynamic Type Definition	
3. Programski jezik Prolog je:			D.
A. Funkcionalni jezik	B. Imperativni jezik	C. Niži programski jezik	
4. Koja je od navedenih XML deklaracija sintaksno ispravna			C.
<ul style="list-style-type: none"> 1. <xml version="1.0" encoding="UTF-8"> 2. <?xml version="1.0" encoding="UTF-8"> 3. <?xml version="1.0" encoding="UTF-8"?> 			
A. 1	B. 3	C. 2	
5. Da li je XML dokument dobro oformljen?			B
<pre><?xml version="1.0" encoding="UTF-8"?> <datum>05.02.05.</datum> <od>Dejan</od> <za>Ana</za> <tekst>Danas je ispit.</tekst></pre>			
A. DA	B. NE		
6. Koja su tvrđenja tačna:			C.
<ul style="list-style-type: none"> 1. Svaki XML dokument mora da ima XML šemu 2. Svaki XML element mora da ima početni i kranjnji tag (oznaku) 3. XML dokument ima jedan i samo jedan koreni element 4. XML elementi nisu case sensitive 			

A. 1 i 2	B. 1 i 3	C. 2 i 3	D. Nijedno	
7. XSLT stylesheet dokument je XML dokument				A.
A. DA	B. NE			
8. Koji od navedenih pristupa se koristi za formalni opis semantike programskog jezika? 1. askiomatski 2. denotacioni 3. kontekstno-slobodne gramatike 4. translacioni				C.
A. 1 i 2	B. 1	C. 1,2,4	D. 2,3,4	
9. Mehanizam sa inkapsulaciju implementacije apstraktnog tipa podatka u C# je				B.
A. Struktura podataka	B. Klasa	C. Package		
10. XML Reader reprezentuje				A.
A. DOM model XML Parsera	B. Pull model XML Parsera	C. SAX model XML parsera		
11. C# kompajler vrši translaciju C# programa u				C.
A. mašinski jezik	B. Bytecode	C. IL (Intermediate Language)		
12. Program napisan u proceduralnom programskom jeziku sastoji se iz:				B.
A. Funkcija	B. Niza instrukcija	C. Logičkih pravila		
13. Imperativna paradigma podržava: 1. Bazne apstrakcije 2. Apstrakcije podataka 3. Proceduralne apstrakcije 4. Strukturne apstrakcije				C.
A. 1,3 i 4	B. 1,2,3 i 4	C. 1, 2 i 4	D. 1 i 4	
14. Specifikacija apstraktnog tipa podataka zadaje se preko: 1. Klase 2. Interfejsa 3. Package 4 Namespace				?
A. 1 i 2	B. 1	C. 1, 2 i 4	D. 2	
15. Bazne apstrakcije u programskim jezicima su: 1. Instrukcije dodeljivanja 2. Predefinisani tipovi podataka 3. Struktore podataka 4. Proceduralne apstrakcije				C.
A. 1, 2 i 3	B. 2	C. 1 i 2	D. 3 i 4	
16. XML šema specificira 1. Skup predefinisanih pravila za opis strukture XML dokumenta 2. Specificira jezik koji se koirsti za transformaciju XML dokumenta 3. XML parser				C.

A. 2	B. 3	C. 1	D. 1 i 2	
17. Atribut match u templejt elementu <xsl:template match="/">				C.
<ol style="list-style-type: none"> 1. Povezuje templejt sa korenim elementom izvornog XML dokumenta 2. Opisuje transformaciju koja se primenjuje na XML dokument 3. Selektuje vrednost XML elementa 				
A. 2 i 3	B. 3	C. 1	D. 1 i 2	
18. Ključna operacija imperativne paradigme je				A.
A. Dodeljivanje	B. Prenosenje poruka	C. Unifikacija		
19. XML je podskup				A.
<ol style="list-style-type: none"> 1. SGML - a 2. HTML - a 3. C# - a 4. Java - e 				
A. 2	B. 4	C. 1	D. 3	
20. Da bi XML dokument bio validan, on mora biti				B
<ol style="list-style-type: none"> 1. Formatiran 2. Konzistentan sa strukturom definisanom u opisu tipa dokumenta 3. Dobro oformljen 				
A. 1 i 3	B. 2 i 3	C. 1 i 3	D. 1,2 i 3	

Ispit 5.2.2005 - Grupa 4

1. Programske paradigme obezbeđuju model projektovanja/implementacije softverskih sistema?				A.
A. DA	B. NE			
2. Koji je ispravan način za referenciranje na stylesheet dokument nazvan "pozdrav.xml"?				C.
<ol style="list-style-type: none"> 1. <stylesheet type="text/xsl" href="pozdrav.xml"/> 2. <link type="text/xsl" href="pozdrav.xml"/> 3. <?xml-stylesheet type="text/xsl" href="pozdrav.xml"?> 				
A. 1	B. 2	C. 3		
3. Karakteristika(e) apstraktnog tipa podataka je (su):				B. i D.
A. Nasleđivanje	B. Učaurjenje	C. Polimorfizam	D. Skrivanje informacija	
4. XML je skraćenica od				C.

A. Example Markup Language	B. eXtensible Markup Language	C. XMarkup Language	
5. Da li je XML dokument dobro оформљен?			A
<pre><?xml version="1.0" encoding="UTF-8"?> <poruka datum>05.02.05.</datum> <od>Dejan</od> <za>Ana</za> <tekst>Danas je ispit.</tekst> </poruka></pre>			
A. DA	B. NE		
6. Koja su tvrđenja tačna:			B.
<ol style="list-style-type: none"> XML dokument ne mora da ima koreni element XML dokument je samoopisjući i platformski nezavisan XML tagovi su predefinisani Dozvoljeno je višestruko ugnježdavanje XML dokumenata 			
A. 1 i 4	B. 2 i 4	C. 2, 3 i 4	D. 1,2, 3 i 4
7. HTML se koristi za			A.
A. formatiranje i prikazivanje podataka		B. opis strukture i semantičkog sadržaja dokumenata	
8. DOM parseri u .NET-u su:			C.
<ol style="list-style-type: none"> MSXML parser XmlDocument XmlTextReader SAX parser 			
A. 1 i 4	B. 1 i 3	C. 1 i 2	D. 2,3 i 4
9. Programski jezik LISP je:			D.
A. Imperativni jezik	B. Mašinski jezik	C. Logički jezik	
10. Koje od navedenih struktura podataka su apstraktni tipovi podataka?			A. i D.
A. Stablo	B. Niz	C. Zapis	
11. U slučaju obrade XML Dokumenta sa prostom strukturom koja sadrži veliki obim podataka i ako se ne zahteva dinamički pristup dokumentima - pogodnije je koristiti			A.
A. DOM parser		B. SAX parser	
12. Koji se od navedenih mehanizama koristi za skrivanje informacija u C#?			C.
A. Package	B. public klauzula (javni interfejs), private klauzula (skrivanje implementacije)	C. Za svaku članicu definiše se vrsta pristupa	
13. Strukturne apstrakcije u programskim jezicima su			D.
<ol style="list-style-type: none"> proceduralne apstrakcije strukture podataka 			

3. apstrakcije podataka 4. upravljачke strukture				
A. 1, 3 i 4	B. 1 i 3	C. 3 i 4	D. 2 i 4	
14. DTD je XML dokument				B
A. DA	B. NE			
15. Da bi XML dokument bio dobro oformljen on mora biti				A.
<ol style="list-style-type: none"> 1. Sintaksno korektan 2. Konzistentan sa strukturom definisanom u opisu tipa dokumenta 3. Formatiran 				
A. 1	B. 2	C. 1 i 3	D. 2 i 3	
16. XSLT procesor za transformaciju XML dokumenta koristi				D.
<ol style="list-style-type: none"> 1. XML šemu 2. HTML dokument 3. XSLT stylesheet dokument 				
A. 1 i 2	B. 2 i 3	C. 1, 2 i 3	D. 1 i 3	
17. Atribut match u templejt elementu <xsl:template match="/">				C.
<ol style="list-style-type: none"> 1. Povezuje templejt sa korenim elementom izvornog XML dokumenta 2. Opisuje transformaciju koja se primenjuje na XML dokument 3. Selektuje vrednost XML elementa 				
A. 2 i 3	B. 3	C. 1	D. 1 i 2	
18. Ključna operacija objektno orijentisane paradigme je				B.
A. Dodeljivanje	B. Prenosenje poruka	C. Unifikacija		
19. Koji se standardi koriste za definisanje tipova XML dokumenata				D.
<ol style="list-style-type: none"> 1. DTD (Document Type Definiton) 2. XSD (XML Schema Definition) 3. XSLT (Extensible Stylesheet Language Transformation) 				
A. 1 i 3	B. 1, 2 i 3	C. 2	D. 1 i 2	
20. SAX model parsira XML dokument i formira memorijsko stablo čvorova koje reprezentuje XML dokument?				B
A. DA	B. NE			