

Öğrenci Numarası : _____ Adı Soyadı : _____

Sınav ,çoktan seçmeli ve klasik olmak üzere iki kısımdan oluşmaktadır. Çoktan seçmeli kısmın cevapları optik okuyucuya uygun cevap kağıtları üzerine işaretlenecektir. İlk **30dk** sonunda optik cevap kağıtları toplanacaktır. Kalan klasik soruları bu süreden sonra cevaplayabilirsiniz. Grubunuzu işaretlemeği unutmayınız!

Soru	1	2	3	4	5	6	Toplam
Puan	50	10	10	10	10	10	100
Not							

1. (1) (2P) Programlama dilleri hakkında aşağıdakilerden hangisi doğrudur?(Which of the following statements about a programming language is correct?)
 - A. Sözdizimi dilin yapısını, semantik ise anlamını ifade eder.(Syntax refers to the structure of the language, while semantics refers to the meaning of the language.)
 - B. Semantik dilin yapısını, sözdizimi anlamını ifade eder.(Semantics refers to the structure of the language, while syntax refers to the meaning of the language.)
 - C. Sözdizimi dilin yapısını, soyutlama anlamını ifade eder.(Syntax refers to the structure of the language, while abstraction refers to the meaning of the language.)
 - D. Semantik dilin yapısını, soyutlama anlamını ifade eder.(Semantics refers to the structure of the language, while abstraction refers to the meaning of the language.)
- (2) (2P) Haskell dilinde lambda sembolü"nu" hangi karakter temsil eder?(In the Haskell language, the character represents the lambda.)
 - A. backslash (\)
 - B. vertical bar (/)
 - C. underscore (_)
 - D. ampersand (&)
- (3) (2P) _____otomatik olarak BNF tanımlamalarını ayrıştırıcılara dönüştürürler.(_____automatically translate a BNF description into a parser.)
 - A.Sözdizimi üreticileri(Syntax generators)
 - B. Tarayıcılar(Scanners)
 - C. Kaydır-azalt ayrıştırıcılar(Shift-reduce parsers)
 - D. Ayrıştırıcı üreticiler(Parser generators)
- (4) (2P) Tip kontrolü"nu" çalışma zamanına ertelemenin getirisi nedir?(The real benefit of deferring type checking until runtime is_.)
 - A.daha yüksek program güvenilirliği(higher program reliability)
 - B. daha iyi programcı verimliliği(greater programmer efficiency)
 - C. daha yüksek program güvenliği(higher program safety)
 - D. daha iyi çalışma zamanı verimliliği(greater runtime efficiency)
- (5) (2P) Global bir değişken aynı isimle yerel bir blok içinde tanımlanırsa ona _____denir.(A variable declared locally within a block is said to _____a global variable of the same name.)
 - A. ayna(mirror)
 - B. etkisiz kılan(override)
 - C. önleyen(occlude)
 - D. gölge(shadow)
- (6) (2P) Dinamik tahsis değişkenleri nereye yerleştirir?(Dynamic allocation places variables _____.)
 - A.sembol tablosuna(in the symbol table)
 - B. yığın alanına(on the stack)
 - C. işaretçi listesine(in the pointer list)
 - D. heap alanına(on the heap)

- (7) (2P) Lisp _____ programlama modeline dayanır. (Lisp is based on the _____ model of computation.)
A. buyurucu(imperative) B. fonksiyonel(functional) C. yorumlamalı(interpretive)
D. anlamsal(semantic)
- (8) (2P) Gramer kuralları _____ olarak da adlandırılır. (Grammar rules are also called _____.)
A. y nerge(instructions) B. belgeleme(documentation) C. u retim(productions)
D. derleme(compilations)
- (9) (2P) Hangisi genellikle y kleme zamanında ba lanır? (_____ are generally bound at load time.)
A.  ntanımı belirlayiciler(Predefined identifiers) B. Global de gi kenler(Global variables)
C. Tu m  zellikler(All attributes) D. Tu m de gi kenler(All variables)
- (10) (2P) Python ba langı ta _____ i in tasarlanmı tır. (Python was originally designed for _____.)
A. geni   l ekli sistemler(large scale systems) B. zamanın  nemli oldu u sistemler(time-critical systems)
C. bilim adamları, mu hendisler ve programcı olmayan di erleri(scientists and engineers and other non-programmers) D. uzman programcılar(expert programmers)
- (11) (2P) De gi kenler, _____ kullanarak bir isim ve tip kazanırlar. (Variables are given names and data types using a_.)
A. yapı(structure) B. fonksiyon(function) C. bildirim(declaration) D. paket(package)
- (12) (2P) Du zenli ifadelerde tekrar _____ sembolu ile g sterilir. (In a regular expression, repetition is indicated by_.)
A. a vertical bar (/) B. an exclamation mark (!) C. an asterisk (*) D. a hyphen (-)
- (13) (2P) C dilinde fonksiyonlardan geri d onu  tipi olarak dizilerin kullanılamaması hangi  zelli in eksik oldu unu g sterir? (The fact that C does not allow array types to be the return value of a function indicates a lack of_in the language.)
A. ortogonallik(orthogonality) B. genellik(generality) C. tekbi imlilik(uniformity)
D. verimlilik(eficiency)
- (14) (2P) Bo  alan karakterlerinin i lenmesi genellikle _____ tarafından ele alınır. (The processing of white space is generally handled by a_.)
A. kestirimci ayrıştırıcı(predictive parser) B. tarayıcı(scanner) C. ayrıştırıcı(parser)
D. ayrıştırıcı u retici(parser generator)
- (15) (2P) A, sa ıdakilerden hangisi kaynak kodu girdi olarak alır ve  alı tırılabilir bir programa d onu tu ur r? (A(n) _____ takes source code as input and translates it into a program that is executable.)
A. ba layıcı(linker) B. y kleyici(loader) C. derleyici(compiler) D. yorumlayıcı(interpreter)
- (16) (2P) E er bir i le  birden fazla fonksiyona kar ılık geliyorsa ona _____ denir. (If an operator performs more than one function it is said to be_.)
A. a ırı y klenmi (overloaded) B.  ortu lu (implicit) C. de gi ken(variable)
D. g lgelenmi (shadowed)

- (17) (2P) Haskell ile ilgili aşağıdakilerden hangisi doğrudur?(Which of the following statements about Haskell is correct?)
A. Haskell tamamıyla curry "ozelliğini desteklemeyen tembel bir dildir.(Haskell is a lazy language that is not fully curried.) B. Haskell tamamıyla curry "ozelliğini destekleyen tembel bir dildir.(Haskell is a fully curried and lazy language.)
C. Haskell tamamıyla curry "ozelliğini destekleyen ve gecikmeli değerlendirilmeye izin vermeyen bir dildir.(Haskell is a fully curried language that does not do delayed evaluation.)
D. Haskell yüksek-dereceli fonksiyonları desteklemez.(Haskell does not support higher-order functions.)
- (18) (2P) C++ sınıf tanımlarından sonra noktalı virgöl gerektirir fakat fonksiyon tanımlarından sonra gerektirmez. Bu "ozelliği ile C++ hangisini sağlamaz?(Because C++ requires a semi-colon after a class definition but not after a function definition, C++ is said to lack_.)
A. güvenilirlik(reliability) B. güvenlik(security) C. tekbiçimlilik(uniformity)
D. ortogonalite(orthogonality)
- (19) (2P) Bir ayrıştırıcının en basit hali bir _____.(The simplest form of a parser is a(n) _____.)
A. yorumlayıcıdır(interpreter) B. çevirmendir(translator) C. tanıyıcıdır(recognizer)
D. derleyicidir(compiler)
- (20) (2P) Aşağıdakilerden hangisi Scheme dilinde bir veri listesini temsil eder?(Which of the following represents a list of data in Scheme?)
A. (quote (3.1, 3.2, 3.3)) B. (3.1, 3.2, 3.3) C. (quote (3.1 3.2 3.3)) D. (3.1; 3.2; 3.3)
- (21) (2P) Aşağıdakilerden hangisi ML dilinde bir veri listesini temsil eder?(Which of the following represents a list in ML?)
A. (1, 2, 3) B. [1, 2, 3] C. {1 2 3} D. (1 2 3)
- (22) (2P) Bir metoda yapılan her çağrı_____olarak adlandırılır.(Each call to a method is referred to as a(n)_____.)
A. dallanma(branching) B. başlatma(invocation) C. yürütme(execution) D. etkinleştirme(activation)
- (23) (2P) Scheme dilinde ((x) (p q) t) listesi üzerinde (car (car (cdr L))) çağrısının sonucu nedir?(Given a list L in Scheme with contents of ((x) (p q) t). What will be returned if the command (car (car (cdr L))) is executed?)
A. (x) B. p C. (p) D. (p q)
- (24) (2P) Diziler aşağıdaki soyutlamalardan hangisidir?(An array is considered to be a _____.)
A.yapısal veri soyutlaması(structured data abstraction) B. basit veri soyutlaması(basic data abstraction) C. basit kontrol soyutlaması(basic control abstraction)
D. birim soyutlaması(unit abstraction)
- (25) (2P) Aşağıdaki dillerin hangisinde statik tip tanımlaması yoktur?(All of the following languages are statically typed except_.)
A. Python B. C C. C++ D. Ada

2. (10P) Parametre olarak kelimelerin virgükle ayrıldığı bir string alan, bu kelimeleri alfabetik olarak sıralayıp yine virgükle ayrıldığı bir string d'öndüren Python metodunu yazınız. Örn: 'kalem,defter,elma,armut,kitap' → 'armut,defter,elma,kalem,kitap'(Write the kelimeSiralama method which takes comma separated numbers as a string, splits and sorts the numbers and returns the combined comma separated numbers as string.)

```
def kelimeSiralama(metin):  
    kelimeler = metin.split(',')  
    kelimeler.sort()  
    return ','.join(kelimeler)
```

3. (10P) "sayilar.txt" isimli dosyada birden fazla satırda aralarında virgüli karakteri bulunacak şekilde tamsayılar birbirinden ayrılmıştır. Bu sayıları dosyadan okuyup her satırda bir sayı olacak şekilde yazdıran Ruby kodunu yazınız.(There are multiple lines in "sayilar.txt" file and in every line there are comma separated numbers. Write a Ruby program that reads numbers from the file and prints a number each line.)

```
File.open('sayilar.txt') do |kayit|  
  kayit.each do |satir|  
    sayilar = satir.split(",")  
    sayilar.each do |deger|  
      puts deger  
    end  
  end  
end
```

4. (10P) Aşağıda bir dizideki en büyük sayıyı bulmak için enbuyuk fonksiyonunun prototipi C dilinde verilmiştir. enbuyuk fonksiyonunu atama ve döngü işlemleri olmadan(mu'mkün olduğunca fonksiyonel) ve kuyruk özyinelemeli olarak yazınız ve main fonksiyonu içinden enbuyuk fonksiyonunun nasıl çağrılacağını gösteriniz.(Below is the prototype of a function which finds the maximum number in an array. Write tail recursive enbuyuk function without using assignment and loops. Also in main, write function call properly.)

```
int enbuyuk(int dizi[], int es, int max)  
{  
    if(es== -1)  
        return max;  
    if(dizi[es]>max)  
        return enbuyuk(dizi, es-1, dizi[es]);  
    else  
        return enbuyuk(dizi, es-1, max);  
}  
void main()  
{  
    int dizi[]={12, 1, 4, 21, 3, 99};  
    printf("%d\n", enbuyuk(dizi, 5, dizi[5]));  
}
```

5. (10P) Aşağıdaki gramerin başlangıç sembolü P'dir. Belirtilen giriş cümlesi verildiğinde oluşan sözdizim ağacını çiziniz. (The start symbol of the grammar below is P. Draw the parse tree using the input string below.)

$P \rightarrow S / S P$

$S \rightarrow SY . K$

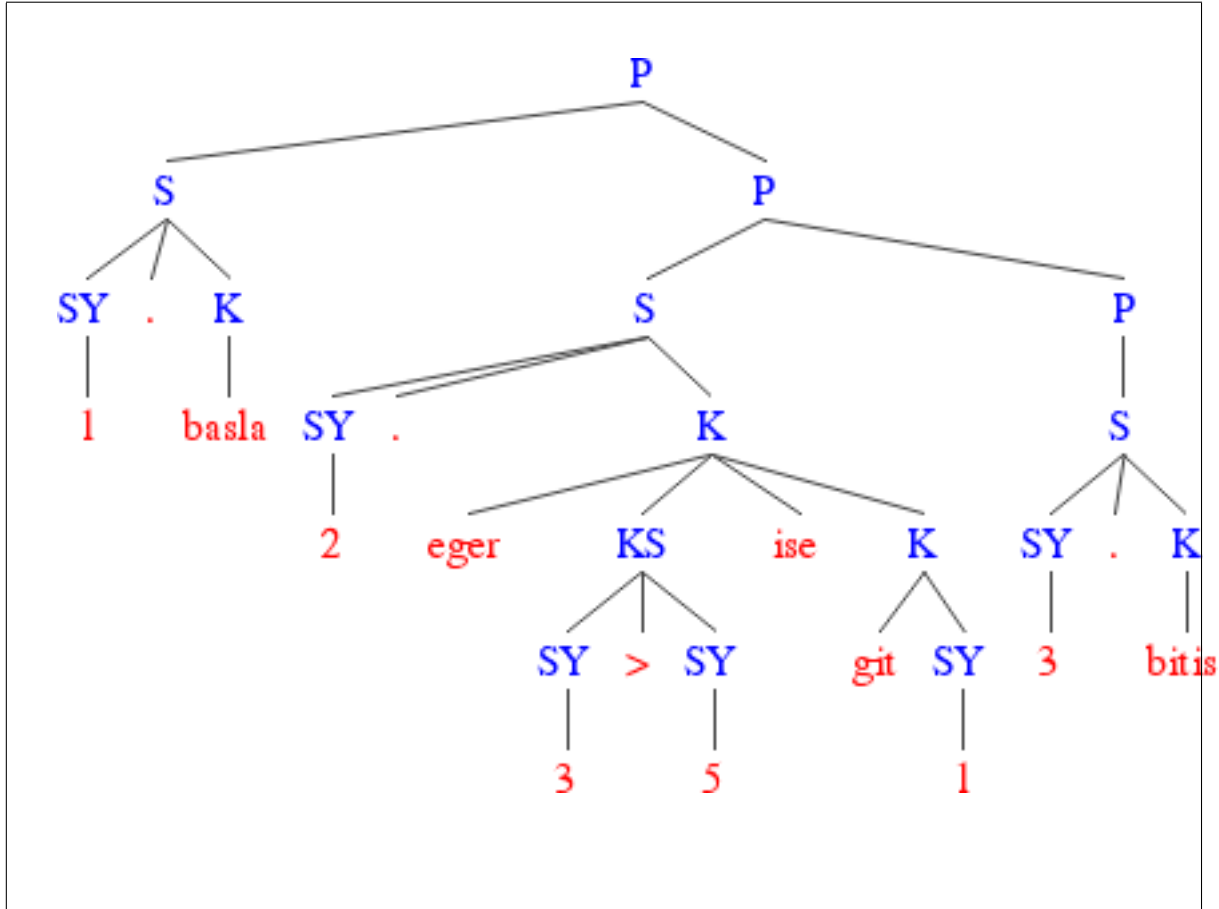
$K \rightarrow yaz\ SY / git\ SY / basla / bitis / eger\ KS\ ise\ K$

$KS \rightarrow SY < SY / SY = SY / SY > SY$

$SY \rightarrow 1 / 2 / 3 / 4 / 5 / 6 / 7 / 8 / 9 / 0$

Giriş cümlesi:

1. basla
2. eger 3>5 ise git 1
3. bitis



6. Aşağıdaki C kodu için kullanılan kapsama göre çıktıları yazınız. (Write the output of the C code below when specific scoping is used.)

```
1 int x;
2 int main() {
3   x = 14;
4   f();
5   g();
6 }
7 void f() {
8   int x = 13;
9   h();
10 }
11 void g() {
12   int x = 12;
13   h();
14 }
15 void h() {
16   printf("%d\n", x);
17 }
```

(1) (5P) Statik kapsam(Lexical scope)

14 14

(2) (5P) Dinamik kapsam(Dynamic scope)

13 12