Matematiksel Fonksiyonlar ve Dizgeler

1. Aşağıdaki kodun çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 int result = Math.max(15, 10) + Math.min(5, 20);  
 System.out.println(result);  
 }  
}

1. 25
2. 30
3. 20
4. 35
5. 15
6. Aşağıdaki kodun boşlukları doğru şekilde dolduran seçenek hangisidir?

public class punchcard {  
 public static void main(String[] args) {  
 x = 15;  
 double result = Math.\_\_\_\_(x);  
 System.out.println(result);  
 }  
}

1. root(x)
2. sqrt(x)
3. sqr(x)
4. squareRoot(x)
5. sqrRoot(x)
6. Aşağıdakilerden hangisi **char** değişkenini doğru şekilde bildirmenin **DOĞRU OLMAYAN** yolu?

public class punchcard {  
 public static void main(String[] args) {  
 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;  
 System.out.println(ch);  
 }  
}

1. char ch = 'A';
2. char ch = 65;
3. char ch = "A";
4. char ch = '\u0041';
5. char ch = 0x41;
6. Terminalde ödül mesajının görüntülenmemesi için boşluğa ne konulmalıdır?

public class punchcard {  
 public static void main(String[] args) {  
 String str = "You found the prize!";  
 System.out.print(str + \_\_\_\_);  
 }  
}

1. "\n"
2. "\t"
3. "\b"
4. "\r"
5. "\f"
6. İkinci çıktı satırındaki ekstra boşlukları nasıl kaldırabilirsiniz? (beklenen çıktı: wooden clocks)

public class punchcard {  
 public static void main(String[] args) {  
 String str = "clock and time";  
 str = str.substring(0, 6);  
 str = "wooden " + str + "\_\_\_\_\_s";  
 System.out.println(str);  
 }  
}

1. "\n"
2. "\t"
3. "\b"
4. "\r"
5. "\f"
6. Aşağıdaki kodun çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 System.out.println("Java\tProgramming\nLanguage");  
 }  
}

A) Java Programming Language

B) Java Programming  
Language

C) Java ProgrammingLanguage

D) Java Programming  
  
Language

E) Java\tProgramming\nLanguage

1. Java’da bir dizenin uzunluğunu elde etmek için hangi yöntem kullanılır?

public class punchcard {  
 public static void main(String[] args) {  
 String str = "The length of this string is 31";  
 System.out.println(str.\_\_\_\_\_\_\_);  
 }  
}

1. len()
2. lengthOf()
3. size()
4. getLength()
5. length()
6. Aşağıdaki kodun çıktısı nedir?

**NOT:** ==, String türündeki referansları karşılaştırmak için kullanılabilir, ancak böyle bir eşitlik testi, iki operatörün aynı String nesnesine işaret edip etmediğini belirler. Operatörler farklı String nesneleri ise sonuç false olur, hatta karakter dizileri aynı karakter dizisini içeriyor olsalar bile (§3.10.5, §3.10.6). İki dizenin s ve t içeriği, s.equals(t) yöntemi çağrısıyla eşitlik için test edilebilir.

// Bu ikisi aynı değere sahip  
new String("test").equals("test") // --> true  
  
// ... ama bunlar aynı nesne değil  
new String("test") == "test" // --> false  
  
// ... bunlar da değil  
new String("test") == new String("test") // --> false  
  
// ... ama bunlar aynı nesne olduğundan  
// derleyici tarafından internedir ve bu nedenle aynı nesneye işaret eder  
"test" == "test" // --> true  
  
// ... dize literalleri derleyici tarafından birleştirilir  
// ve sonuçlar interned olur.  
"test" == "te" + "st" // --> true

public class punchcard {  
 public static void main(String[] args) {  
 String first = "Hello";  
 String second = "Hello";  
 System.out.println(first == second);  
 }  
}

1. true
2. false
3. Derleme zamanı hatası
4. Çalışma zamanı istisnası
5. NullPointerException
6. Aşağıdaki kodun çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String first = "H";  
 String second = 'H';  
 System.out.println(first.equals(second));  
 }  
}

1. true
2. false
3. Derleme zamanı hatası
4. Çalışma zamanı istisnası
5. NullPointerException
6. Aşağıdaki kodun çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String word1 = "Java";  
 String word2 = "JavaIsFun";  
 System.out.println(word1.compareTo(word2));  
 }  
}

1. 0
2. Negatif bir tamsayı
3. Pozitif bir tamsayı
4. True
5. False
6. Aşağıdaki kod parçasının çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String original = "brown fox";  
 String substring = original.substring(2, 8);  
 System.out.println(substring);  
 }  
}

1. brown
2. wn fox
3. own fox
4. own fo
5. rown f
6. Aşağıdaki kod parçasının çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 if ( 'A' + "BC" == "AB" + 'C' ) {  
 System.out.println("Equal");  
 } else {  
 System.out.println("Not Equal");  
 }  
 }  
}

1. Equal
2. Not Equal
3. hata çünkü char dizeyle birleştirilemez
4. hata çünkü dizeler == ile karşılaştırılamaz
5. hata çünkü operatörler aynı türde değil
6. Aşağıdaki kod parçasının çıktısı nedir?

Kullanıcı girişinin hem input1 hem de input2 için ‘clock’ olduğunu varsayalım.

import java.util.Scanner;  
  
public class punchcard {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
 String input1 = sc.next();  
 String input2 = sc.next();  
 System.out.println(input1 == input2);  
 }  
}

1. true
2. false
3. Derleme zamanı hatası
4. Çalışma zamanı istisnası
5. NullPointerException
6. Aşağıdaki kod parçasının çıktısı nedir?

import java.util.Scanner;  
  
public class punchcard {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
 String input1 = sc.next();  
 String input2 = input1;  
 System.out.println(input1 == input2);  
 }  
}

1. true
2. false
3. Derleme zamanı hatası
4. Çalışma zamanı istisnası
5. NullPointerException
6. Aşağıdaki kod parçasının çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String sentence = "Java is fun to learn";  
 int index = sentence.indexOf('a');  
 System.out.println(index);  
 }  
}

1. 0
2. 1
3. 2
4. 1 3 16
5. 2 4 17
6. Aşağıdaki kod parçasının çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String sentence = "Java is fun to learn";  
 int index = sentence.indexOf('j');  
 System.out.println(index);  
 }  
}

1. -1
2. 0
3. False
4. derleme zamanı hatası
5. çalışma zamanı istisnası
6. Aşağıdaki kod parçasının çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String sentence = "Java is fun to learn";  
 int index = sentence.indexOf('a', 3);  
 System.out.println(index);  
 }  
}

1. -1
2. 1
3. 3
4. 17
5. derleme zamanı hatası
6. Aşağıdaki kod parçasının çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String first = "Hello";  
 String second = "World";  
 String result = 1 + 2 + first + second + 20 + 23;  
 System.out.println(result);  
 }  
}

1. 3HelloWorld2023
2. 12HelloWorld2023
3. 3HelloWorld43
4. 12HelloWorld43
5. Derleme zamanı hatası
6. Aşağıdaki kod parçasının çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String first = "lazy";  
 String second = "dog";  
 String result = first.toUpperCase() + " " + second.toLowerCase();  
 System.out.print(result);  
 String result2 = (first.toUpperCase()).concat(second.toLowerCase());  
 System.out.println(result2);  
 }  
}

1. LAZY dogLAZYdog
2. LAZY dogLAZY dog
3. LAZYdogLAZYdog
4. LAZYdogLAZY dog
5. Derleme zamanı hatası
6. Aşağıdaki kod parçasının çıktısı nedir?

public class punchcard {  
 public static void main(String[] args) {  
 String sentence = "Java is fun to learn";  
 int index = sentence.indexOf("fun");  
 System.out.println(index);  
 }  
}

1. -1
2. 7
3. 8
4. 9
5. derleme zamanı hatası

*Cevaplar:*

1. C - 20
2. B - sqrt(x)
3. C - char ch = "A";
4. A - "\r"
5. C - "\b"
6. B
7. E - length()
8. A - true
9. C - Compile-time error
10. B - a negative integer
11. D - own fo
12. A - Equal
13. B - false
14. A - true
15. B - 1
16. A - -1
17. D - 17
18. A - 3HelloWorld2023
19. A - LAZY dogLAZYdog
20. C - 8