

STANDARD CLASSES AND INTERFACES – SUPPLEMENTAL REFERENCE

package java.lang

```
class Object
    boolean equals(Object anotherObject)
    String toString()
    int hashCode()

interface Comparable<T>
    int compareTo(T anotherObject)
        Returns a value < 0 if this is less than anotherObject.
        Returns a value = 0 if this is equal to anotherObject.
        Returns a value > 0 if this is greater than anotherObject.

class Integer implements Comparable<Integer>
    Integer(int value)
    int intValue()
    boolean equals(Object anotherObject)
    String toString()
    String toString(int i, int radix)
    int compareTo(Integer anotherInteger)
    static int parseInt(String s)

class Double implements Comparable<Double>
    Double(double value)
    double doubleValue()
    boolean equals(Object anotherObject)
    String toString()
    int compareTo(Double anotherDouble)
    static double parseDouble(String s)

class String implements Comparable<String>
    int compareTo(String anotherString)
    boolean equals(Object anotherObject)
    int length()
    String substring(int begin)
        Returns substring(begin, length()).
    String substring(int begin, int end)
        Returns the substring from index begin through index (end - 1).
    int indexOf(String str)
        Returns the index within this string of the first occurrence of str.
        Returns -1 if str is not found.
    int indexOf(String str, int fromIndex)
        Returns the index within this string of the first occurrence of str,
        starting the search at fromIndex. Returns -1 if str is not found.
    int indexOf(int ch)
    int indexOf(int ch, int fromIndex)
    char charAt(int index)
    String toLowerCase()
    String toUpperCase()
    String[] split(String regex)
    boolean matches(String regex)
    String replaceAll(String regex, String str)

class Character
    static boolean isDigit(char ch)
    static boolean isLetter(char ch)
    static boolean isLetterOrDigit(char ch)
    static boolean isLowerCase(char ch)
    static boolean isUpperCase(char ch)
    static char toUpperCase(char ch)
    static char toLowerCase(char ch)

class Math
    static int abs(int a)
    static double abs(double a)
    static double pow(double base, double exponent)
    static double sqrt(double a)
    static double ceil(double a)
    static double floor(double a)
    static double min(double a, double b)
    static double max(double a, double b)
    static int min(int a, int b)
    static int max(int a, int b)
    static long round(double a)
    static double random()
        Returns a double greater than or equal to 0.0 and less than 1.0.
```

package java.util

```
interface List<E>
class ArrayList<E> implements List<E>
    boolean add(E item)
    int size()
    Iterator<E> iterator()
    ListIterator<E> listIterator()
    E get(int index)
    E set(int index, E item)
    void add(int index, E item)
    E remove(int index)

class LinkedList<E> implements List<E>, Queue<E>
    void addFirst(E item)
    void addLast(E item)
    E getFirst()
    E getLast()
    E removeFirst()
    E removeLast()

class Stack<E>
    boolean isEmpty()
    E peek()
    E pop()
    E push(E item)

interface Queue<E>
class PriorityQueue<E>
    boolean add(E item)
    boolean isEmpty()
    E peek()
    E remove()

interface Set<E>
class HashSet<E> implements Set<E>
class TreeSet<E> implements Set<E>
    boolean add(E item)
    boolean contains(Object item)
    boolean remove(Object item)
    int size()
    Iterator<E> iterator()
    boolean addAll(Collection<? extends E> c)
    boolean removeAll(Collection<?> c)
    boolean retainAll(Collection<?> c)

interface Map<K,V>
class HashMap<K,V> implements Map<K,V>
class TreeMap<K,V> implements Map<K,V>
    Object put(K key, V value)
    V get(Object key)
    boolean containsKey(Object key)
    int size()
    Set<K> keySet()
    Set<Map.Entry<K, V>> entrySet()

interface Iterator<E>
    boolean hasNext()
    E next()
    void remove()

interface ListIterator<E> extends Iterator<E>
    void add(E item)
    void set(E item)

class Scanner
    Scanner(InputStream source)
    Scanner(String str)
    boolean hasNext()
    boolean hasNextInt()
    boolean hasNextDouble()
    String next()
    int nextInt()
    double nextDouble()
    String nextLine()
    Scanner useDelimiter(String regex)
```

The time limit for the contest is 45 minutes. Each correct question is awarded 6 points; no points are given or subtracted if unanswered; 2 points are deducted for an incorrect answer.

Best of Texas 2025-26 Computer Science Test 1

1. $25_8 * 3_8 =$

- a. 63_8
- b. 77_8
- c. 54_8
- d. 65_8
- e. 73_8

2. What is the output of the code segment at right?

- a. 200
- b. 40
- c. 50
- d. 11

```
out.print((34 + 6)*(11 % 5));
```

3. What is the output of the code segment at right?

- a. Red
BlueGreen
Yellow
- b. Red
Blue
Green
Yellow
- c. Red
BlueGreenYellow
- d. Red\nBlueGreen
Yellow

```
System.out.print("Red");  
System.out.print("\nBlue");  
System.out.println("Green");  
System.out.print("Yellow");
```

4. What is the output of the code segment at right?

- a. animal
- b. imal
- c. lami
- d. 4

```
String s1 = "animal";  
String s2 = "al";  
out.print(s1.indexOf(s2,1));
```

5. What is the output of the code segment at right?

- a. 2
- b. 3
- c. 4
- d. 5

```
int[] ar = {1,2,3,4,5};  
int i = 1; arr  
value = ar[++i];  
out.println(num);  
          value
```

6. What is the output of the code segment at right?

- a. 22
- b. 121
- c. 212
- d. There is no output.

```
int x = 11;  
if (x >= 10)  
    if (x >= 20)  
  
out.print(Math.pow(x,2));  
else  
    out.print(x * 2);
```

7. What is the output of the code segment at right?

- a. 6
- b. 8
- c. 9
- d. 11

```
String s = "puppy dog";  
int length = s.length();  
out.print(length);
```

8. What is the output of the code segment at right?

- a. Passed
- b. End of program
- c. Passed
End of program
- d. There is no output.

```
int score = 85;  
  
if (score >= 70) {  
    System.out.println("Passed");  
}  
  
System.out.println("End of program");
```

9. What is the output of the code segment at right?

- a. Hello Hello Hello
- b. HelloHelloHello
- c. Hello Hello Hello Hello
- d. Hello
Hello

```
int x = 0;  
  
while (x < 3) {  
    System.out.print("Hello ");  
    x++;  
}
```


10. What is the value of a following execution of the code segment at right?

- a. 2
- b. 3
- c. 4
- d. 5

```
int a, b = 3, c = 4;  
a = ++b < ++c ? ++b : ++c;
```

11. What is the value of a following execution of the code segment at right?

- a. 1
- b. 2
- c. 12
- d. -1

```
int a = 12; int b = 15;  
while (b >= 0)  
{  
    a = a-- ;  
    b = --b ;  
}
```

12. What is the value of a following execution of the code segment at right?

- a. 15
- b. 1
- c. 0
- d. -1

13. What is the output of the code segment at right?

- a. 2
- b. 5
- c. 7
- d. error

```
int[] ar = {3,5,7,9};  
out.print(ar.get(2));
```

14. What is the output of the code segment at right?

- a. 3
- b. 4
- c. 5
- d. 6

```
double d = 3.5;  
int i = (int) (++d);  
out.println(i);
```