**Review - MCQ**

**Review 1**

**Binary & decimal & octal & hexadecimal**

0.A common use of hexadecimal numerals is to specify colors on web pages. Every color has a red, green, and blue component. In decimal notation, these are denoted with an ordered triple (x, y, z), where x, y, and z are the three components, each an int from 0 to 255. For example, a certain shade of red, whose red, green, and blue components are 238, 9, and 63, is represented as (238,9,63).

In hexadecimal, a color is represented in the format #RRGGBB, where RR, GG, and BB are hex values for the red, green, and blue. Using this notation, the color (238,9,63) would be coded as #EE093F.

Which of the following hex codes represents the color (14,20,255)?

(A) #1418FE

(B) #0E20FE

(C) #0E14FF

(D) #0FE5FE

(E) #0D14FF

1、Suppose that base-2 (binary) numbers and base-16 (hexadecimal) numbers can be denoted with subscripts, as shown below:

2Ahex = 101010bin

Which is equal to 3Dhex?

(A) 111101bin

(B) 101111bin

(C) 10011bin

(D) 110100bin

(E) 101101bin

**Print method & String literal**

2、Consider the following code segment.

System.out.print(\*);     // Line 1

System.out.print("\*");   // Line 2

System.out.println();    // Line 3

System.out.println("\*"); // Line 4

The code segment is intended to produce the following output, but may not work as intended.

\*\*

\*

Which line of code, if any, causes an error?

A Line 1

B Line 2

C Line 3

D Line 4

E The code segment works as intended.

3.Consider the code segment below.

int a = 1988;

int b = 1990;

String claim = " that the world’s athletes " +

"competed in Olympic Games in ";

String s = "It is " + true + claim + a +

" but " + false + claim + b + ".";

System.out.println(s);

What, if anything, is printed when the code segment is executed?

A It is true claima but false claim b.

B It is true claim1998 but false claim1990.

C It is true that the world’s athletes competed in Olympic Games in a but false that the world’s athletes competed in Olympic Games in b.

D It is true that the world’s athletes competed in Olympic Games in 1988 but false that the world’s athletes competed in Olympic Games in 1990.

E Nothing is printed because the code segment does not compile.

**Operator**

3.Consider the following code segment.

int a = 5;

int b = 4;

int c = 2;

a \*= 3;

b += a;

b /= c;

System.out.print(b);

What is printed when the code segment is executed?

A 2

B 4

C 9

D 9.5

E 19

**Casting & Operator procedure**

4.Consider the following code segment.

double x = (int) (5.5 - 2.5);

double y = (int) 5.5 - 2.5;

System.out.println(x - y);

What is printed as a result of executing the code segment?

A -1.0

B -0.5

C 0.0

D 0.5

E 1.0

9.Refer to the following code fragment:

double answer = 13 / 5;

System.out.println("13 / 5 = " + answer);

The output is

13 / 5 = 2.0

The programmer intends the output to be

13 / 5 = 2.6

Which of the following replacements for the first line of code will not fix the problem?

(A) double answer = (double) 13 / 5;

(B) double answer = 13 / (double) 5;

(C) double answer = 13.0 / 5;

(D) double answer = 13 / 5.0;

(E) double answer = (double) (13 / 5);

**Data type**

1. Consider the following code segment.

int x = 5;

int y = 6;

/\* missing code \*/

z = (x + y) / 2;

Which of the following can be used to replace /\* missing code \*/ so that the code segment will compile?

**I** int z = 0;

**II** int z;

**III** boolean z = false;

A I only

B II only

C I and II only

D II and III only

E I, II, and III

**Double-round-off error**

7.Which is true of the following boolean expression, given that x is a variable of type double?

3.0 == x \* (3.0 / x)

(A) It will always evaluate to false.

(B) It may evaluate to false for some values of x.

(C) It will evaluate to false only when x is zero.

(D) It will evaluate to false only when x is very large or very close to zero.

(E) It will always evaluate to true.

**Review 2**

**Boolean expressions**

1、Consider the following Boolean expression in which the int variables x and y have been properly declared and initialized.

(x <= 10) == (y > 25)

Which of the following values for x and y will result in the expression evaluating to true ?

A x = 8 and y = 25

B x = 10 and y = 10

C x = 10 and y = 30

D x = 15 and y = 30

E x = 25 and y = 30

**Boolean expressions &Logical operator**

2、Consider the following code segment.

if (false && true || false)

{

if (false || true && false)

{

System.out.print("First");

}

else

{

System.out.print("Second");

}

}

if (true || true && false)

{

System.out.print("Third");

}

What is printed as a result of executing the code segment?

A First

B Second

C Third

D FirstThird

E SecondThird

3、Consider the following code segment.

if (a < b || c != d)

{

System.out.println("dog");

}

else

{

System.out.println("cat");

}

Assume that the int variables a, b, c, and d have been properly declared and initialized. Which of the following code segments produces the same output as the given code segment for all values of a, b, c, and d ?

A if (a < b && c != d)

{

System.out.println("dog");

}

else

{

System.out.println("cat");

}

B if (a < b && c != d)

{

System.out.println("cat");

}

else

{

System.out.println("dog");

}

C if (a > b && c == d)

{

System.out.println("cat");

}

else

{

System.out.println("dog");

}

D if (a >= b || c == d)

{

System.out.println("cat");

}

else

{

System.out.println("dog");

}

E if (a >= b && c == d)

{

System.out.println("cat");

}

else

{

System.out.println("dog");

}

**Different between if-else if-else and if-if-else**

4.Consider the following code segments, which are each intended to convert grades from a 100-point scale to a 4.0-point scale and print the result. A grade of 90 or above should yield a 4.0, a grade of 80 to 89 should yield a 3.0, a grade of 70 to 79 should yield a 2.0, and any grade lower than 70 should yield a 0.0.

Assume that grade is an int variable that has been properly declared and initialized.

Code Segment I

double points = 0.0;

if (grade > 89)

{

points += 4.0;

}

else if (grade > 79)

{

points += 3.0;

}

else if (grade > 69)

{

points += 2.0;

}

else

{

points += 0.0;

}

System.out.println(points);

Code Segment II

double points = 0.0;

if (grade > 89)

{

points += 4.0;

}

if (grade > 79)

{

grade += 3.0;

}

if (grade > 69)

{

points += 2.0;

}

if (grade < 70)

{

points += 0.0;

}

System.out.println(points);

Which of the following statements correctly compares the values printed by the two methods?

A The two code segments print the same value only when grade is below 80.

B The two code segments print the same value only when grade is 90 or above or grade is below 80.

C The two code segments print the same value only when grade is 90 or above.

D Both code segments print the same value for all possible values of grade.

E The two code segments print different values for all possible values of grade.

**Iteration -while**

5.Consider the following code segment.

int count = 5;

while (count < 100)

{

count = count \* 2;

}

count = count + 1;

What will be the value of count as a result of executing the code segment?

A 100

B 101

C 160

D 161

E 321

**Iteration -nested for loop**

6.Consider the following code segment.

int counter = 0;

for (int x = 10; x > 0; x--)

{

for (int y = x; y <= x; y++)

{

counter++;  // line 6

}

}

How many times will the statement in line 6 be executed as a result of running the code segment?

A 0

B 1

C 10

D 11

E 20

**Review 3**

**Objects & classes**

1、A student has created a Song class. The class contains the following variables.

A String variable called artist to represent the artist name

A String variable called title to represent the song title

A String variable called album to represent the album title

The object happyBirthday will be declared as type Song.

Which of the following statements is true?

(A) artist, title, and album are instances of the Song class.

(B) happyBirthday is an instance of three String objects.

(C) happyBirthday is an instance of the Song class.

(D) Song is an instance of the happyBirthday object.

(E) Song is an instance of three String objects.

**Calling method**

2.Consider the following class definition.

public class Thing

{

public void talk()

{

System.out.print("Hello ");

}

public void name()

{

System.out.print("my friend");

}

public void greet()

{

talk();

name();

}

/\* Constructors not shown \*/

}

Which of the following code segments, if located in a method in a class other than Thing, will cause the message

"Hello my friend" to be printed?

(A) Thing a = new Thing();

Thing.talk();

Thing.name();

(B) Thing a = new Thing();

Thing.greet();

(C) Thing a = new Thing();

a.talk();

(D) Thing a = new Thing();

a.greet();

(E) Thing a = new Thing();

a.name();

a.talk();

3、Consider the following methods, which appear in the same class.

public void printSum(int x, double y)

{

System.out.println(x + y);

}

public void printProduct(double x, int y)

{

System.out.println(x \* y);

}

Consider the following code segment, which appears in a method in the same class as printSum and

printProduct.

int num1 = 5;

double num2 = 10.0;

printSum(num1, num2);

printProduct(num1, num2);

What, if anything, is printed as a result of executing the code segment?

(A) 15

50

(B) 15

50.0

(C) 15.0

50

(D) 15.0

50.0

(E) Nothing is printed because the code does not compile.

**Overload method**

4、Consider the following attempts at method overloading.

**I.** public class Overload

{

public int average(int x, int y)

{ /\* implementation not shown \*/ }

public int average(int value1, int value2)

{ /\* implementation not shown \*/ }

// There may be instance variables, constructors,

// and methods that are not shown.

}

**II.** public class Overload

{

public int average(int x, int y)

{ /\* implementation not shown \*/ }

public int average(int x, int y, int z)

{ /\* implementation not shown \*/ }

// There may be instance variables, constructors

// and methods that are not shown.

}

**III.** public class Overload

{

public int average(int x, int y)

{ /\* implementation not shown \*/ }

public int average(double x, double y)

{ /\* implementation not shown \*/ }

// There may be instance variables, constructors,

// and methods that are not shown.

}

Which of the attempts at method overloading will compile without error?

(A) I only

(B) II only

(C) III only

(D) II and III only

(E) I, II, and III

**Aliases**

Consider the following class declaration

public class SomeClass

{

  private int num;

  public SomeClass(int n)

  {

    num = n;

  }

  public void increment(int more)

  {

    num = num + more;

  }

  public int getNum()

  {

    return num;

  }

}

The following code segment appears in another class.

SomeClass one = new SomeClass(100);

SomeClass two = new SomeClass(100);

SomeClass three = one;

one.increment(200);

System.out.println(one.getNum() + " " + two.getNum() + " " +

                   three.getNum());

What is printed as a result of executing the code segment?

A 100 100 100

B 300 100 100

C 300 100 300

D 300 300 100

E 300 300 300

**Review-4**

**String class method**

1、Consider the following method.

/\*\* Precondition: Strings one and two have the same length. \*/

public static String combine(String one, String two)

{

String res = "";

for (int k = 0; k < one.length(); k++)

{

if (one.substring(k, k + 1).equals(two.substring(k, k + 1)))

{

res += one.substring(k, k + 1);

}

else

{

res += "0";

}

}

return res;

}

What is returned as a result of the call combine("10110", "01100") ?

A "00000"

B "00100"

C "00101"

D "10110"

E "11011"

2、Consider the following method, which is intended to count the number of times the letter "A" appears in the string str.

public static int countA(String str)

{

int count = 0;

while (str.length() > 0)

{

int pos = str.indexOf("A");

if (pos >= 0)

{

count++;

/\* missing code \*/

}

else

{

return count;

}

}

return count;

}

Which of the following should be used to replace /\* missing code \*/ so that method countA will work as intended?

A str = str.substring(0, pos);

B str = str.substring(0, pos + 1);

C str = str.substring(pos - 1);

D str = str.substring(pos);

E str = str.substring(pos + 1);

**equals() & ==**

Consider the following code segment.

String alpha = new String("APCS");

String beta = new String("APCS");

String delta = alpha;

System.out.println(alpha.equals(beta));

System.out.println(alpha == beta);

System.out.println(alpha == delta);

What is printed as a result of executing the code segment?

(A) false

false

false

(B) false

false

true

(C) true

false

false

(D) true

false

true

(E) true

true

true

**Wrapper class**

2、Consider the following code segment.

double d1 = 10.0;

Double d2 = 20.0;

Double d3 = new Double(30.0);

double d4 = new Double(40.0);

System.out.println(d1 + d2 + d3.doubleValue() + d4);

What, if anything, is printed when the code segment is executed?

(A) 100.0

(B) 10.050.040.0

(C) 10.020.070.0

(D) 10.020.030.040.0

(E) There is no output due to a compilation error.

**Math class**

**1、**Assume that the following variable declarations have been made.

double d = Math.random();

double r;

Which of the following assigns a value to r from the uniform distribution over the range 0.5 ≤ r < 5.5 ?

A r = d + 0.5

B r = d + 0.5 \* 5.0;

C r = d \* 5.0;

D r = d \* 5.0 + 0.5;

E r = d \* 5.5;

1. Consider the following method, which is intended to calculate and return the expression

public double calculate(double x, double y, double a, double b)

{

return /\* missing code \*/;

}

Which of the following can replace /\* missing code \*/ so that the method works as intended?

A Math.sqrt(x ^ 2, y ^ 2, a - b)

B Math.sqrt((x + y) ^ 2 / Math.abs(a - b))

C Math.sqrt((x + y) ^ 2 / Math.abs(a - b))

D Math.sqrt(Math.pow(x + y, 2) / Math.abs(a, b))

E Math.sqrt(Math.pow(x + y, 2) / Math.abs(a - b))

**Review 5**

**Local variable**

1、Consider the following class declaration.

public class Circle

{

private double radius;

public double computeArea()

{

private double pi = 3.14159;

public double area = pi \* radius \* radius;

return area;

}

// Constructor not shown.

}

Which of the following best explains why the computeArea method will cause a compilation error?

(A) Local variables declared inside a method cannot be declared as public or private.

(B) Local variables declared inside a method must all be private.

(C) Local variables declared inside a method must all be public.

(D) Local variables used inside a method must be declared at the end of the method.

(E) Local variables used inside a method must be declared before the method header.

2、The class Worker is defined below. The class includes the method getEarnings, which is intended to return the total amount earned by the worker.

public class Worker

{

private double hourlyRate;

private double hoursWorked;

private double earnings;

public Worker(double rate, double hours)

{

hourlyRate = rate;

hoursWorked = hours;

}

private void calculateEarnings()

{

double earnings = 0.0;

earnings += hourlyRate \* hoursWorked;

}

public double getEarnings()

{

calculateEarnings();

return earnings;

}

}

The following code segment appears in a method in a class other than Worker. The code segment is intended to print the value 800.0, but instead prints a different value because of an error in the Worker class.

Worker bob = new Worker(20.0, 40.0);

System.out.println(bob.getEarnings());

Which of the following best explains why an incorrect value is printed?

(A) The private variables hourlyRate and hoursWorked are not properly initialized.

(B) The private variables hourlyRate and hoursWorked should have been declared

public.

(C) The private method calculateEarnings should have been declared public.

(D) The variable earnings in the calculateEarnings method is a local variable.

(E) The variables hourlyRate and hoursWorked in the calculateEarnings method are

local variables.

**Static**

1.Here are the private instance variables for a Frog object:

public class Frog

{

private String species;

private int age;

private double weight;

private Position position; //position (x,y) in pond

private boolean amAlive;

...

Which of the following methods in the Frog class is the best candidate for being a static method?

A swim //frog swims to new position in pond

B getPondTemperature //returns temperature of pond

C eat //frog eats and gains weight

D getWeight //returns weight of frog

E die //frog dies with some probability based

//on frog’s age and pond temperature

2.Consider the following class definition.

public class Gadget

{

private static int status = 0;

public Gadget()

{

status = 10;

}

public static void setStatus(int s)

{

status = s;

}

}

The following code segment appears in a method in a class other than Gadget.

Gadget a = new Gadget();

Gadget.setStatus(3);

Gadget b = new Gadget();

Which of the following best describes the behavior of the code segment?

A The code segment does not compile because the setStatus method should be called on an object of the class Gadget, not on the class itself.

B The code segment does not compile because the static variable status is not properly initialized.

C The code segment creates two Gadget objects a and b. The class Gadget’s static variable status is set to 10, then to 3, and then back to 10.

D The code segment creates two Gadget objects a and b. After executing the code segment, the object a has a status value of 3 and the object b has a status value of 3.

E The code segment creates two Gadget objects a and b. After executing the code segment, the object a has a status value of 3 and the object b has a status value of 10.

**Override methods**

Consider the following class definitions.

public class Apple

{

public void printColor()

{

System.out.print("Red");

}

}

public class GrannySmith extends Apple

{

public void printColor()

{

System.out.print("Green");

}

}

public class Jonagold extends Apple

{

// no methods defined

}

The following statement appears in a method in another class.

someApple.printColor();

Under which of the following conditions will the statement print "Red" ?

I When someApple is an object of type Apple

II When someApple is an object of type GrannySmith

III When someApple is an object of type Jonagold

A I only

B I and II only

C I and III only

D II and III only

E I, II, and III

**Super keyword**

1.Consider the following class definitions.

public class Book

{

private String author;

private String title;

public Book(String the\_author, String the\_title)

{

author = the\_author;

title = the\_title;

}

}

public class Textbook extends Book

{

private String subject;

public Textbook(String the\_author, String the\_title, String the\_subject)

{

/\* missing implementation \*/

}

}

Which of the following can be used to replace /\* missing implementation \*/ so that the Textbook constructor compiles without error?

A author = the\_author;

title = the\_title;

subject = the\_subject;

B super(the\_author, the\_title);

super(the\_subject);

C subject = the\_subject;

super(the\_author, the\_title);

D super(the\_author, the\_title);

subject = the\_subject;

E super(the\_author, the\_title, the\_subject);

3.Consider the following class definitions.

public class Artifact

{

private String title;

private int year;

public Artifact(String t, int y)

{

title = t;

year = y;

}

public void printInfo()

{

System.out.print(title + " (" + year + ")");

}

}

public class Artwork extends Artifact

{

private String artist;

public Artwork(String t, int y, String a)

{

super(t, y);

artist = a;

}

public void printInfo()

{

/\* missing implementation \*/

}

}

The following code segment appears in a method in another class.

Artwork starry = new Artwork("The Starry Night", 1889, "Van Gogh");

starry.printInfo();

The code segment is intended to produce the following output.

The Starry Night (1889) by Van Gogh

Which of the following can be used to replace /\* missing implementation \*/ in the printInfo method in the Artwork class so that the code segment produces the intended output?

A System.out.print(title + " (" + year + ") by " + artist);

B super.printInfo(artist);

C System.out.print(super.printInfo() + " by " + artist);

D super();

System.out.print(" by " + artist);

E super.printInfo();

System.out.print(" by " + artist);

**Creating references using inheritance hierarchies**

1.Consider the following class definitions.

public class Game

{

private String name;

public Game(String n)

{

name = n;

}

// Rest of definition not shown

}

public class BoardGame extends Game

{

public BoardGame(String n)

{

super(n);

}

// Rest of definition not shown

}

The following code segment appears in a class other than Game or BoardGame.

Game g1 = new BoardGame("checkers");

BoardGame g2 = new Game("chess");

ArrayList<Game> My\_Games = new ArrayList();

My\_Games.add(g1);

My\_Games.add(g2);

Which of the following best explains why the code segment does not compile?

A A BoardGame object cannot be assigned to the Game reference g1.

B A Game object cannot be assigned to the BoardGame reference g2.

C The My\_Games object cannot contain elements of different types.

D The object referenced by g1 cannot be added to My\_Games since g1 was instantiated by a call to the BoardGame constructor.

E The object referenced by g2 cannot be added to My\_Games since g2 was declared to be of type BoardGame.

**Polymorphism**

2、Consider the following class definitions.

public class C1

{

public C1()

{ /\* implementation not shown \*/ }

public void m1()

{ System.out.print("A"); }

public void m2()

{ System.out.print("B"); }

}

public class C2 extends C1

{

public C2()

{ /\* implementation not shown \*/ }

public void m2()

{ System.out.print("C"); }

}

The following code segment appears in a class other than C1 or C2.

C1 obj1 = new C2();

obj1.m1();

obj1.m2();

The code segment is intended to produce the output AB. Which of the following best explains why the code segment does not produce the intended output?

A A compile-time error occurs because obj1 is declared as type C1 but instantiated as type C2.

B A runtime error occurs because method m1 does not appear in C2.

C Method m1 is not executed because it does not appear in C2.

D Method m2 is executed from the subclass instead of the superclass because obj1 is instantiated as a C2 object.

E Method m2 is executed twice (once in the subclass and once in the superclass) because it appears in both classes.

3、Consider the following two class definitions.

public class Bike

{

private int numOfWheels = 2;

public int getNumOfWheels()

{

return numOfWheels;

}

}

public class EBike extends Bike

{

private int numOfWatts;

public EBike(int watts)

{

numOfWatts = watts;

}

public int getNumOfWatts()

{

return numOfWatts;

}

}

The following code segment occurs in a class other than Bike or EBike.

Bike b = new EBike(250);

System.out.println(b.getNumOfWatts());

System.out.println(b.getNumOfWheels());

Which of the following best explains why the code segment does not compile?

A The Bike superclass does not have a constructor.

B There are too many arguments to the EBike constructor call in the code segment.

C The first line of the subclass constructor is not a call to the superclass constructor.

D The getNumOfWatts method is not found in the Bike class.

E The getNumOfWheels method is not found in the EBike class.

**Object subclass （toString() & equals()）**

1、Consider the following class definition.

public class Beverage

{

private int temperature;

public Beverage(int t)

{

temperature = t;

}

public int getTemperature()

{

return temperature;

}

public boolean equals(Object other)

{

if (other == null)

{

return false;

}

Beverage b = (Beverage) other;

return (b.getTemperature() == temperature);

}

}

The following code segment appears in a class other than Beverage. Assume that x and y are properly declared and initialized int variables.

Beverage hotChocolate = new Beverage(x);

Beverage coffee = new Beverage(y);

boolean same = /\* missing code \*/;

Which of the following can be used as a replacement for /\* missing code \*/ so that the boolean variable same is set to true if and only if the hotChocolate and coffee objects have the same temperature values?

A (hotChocolate = coffee)

B (hotChocolate == coffee)

C hotChocolate.equals(coffee)

D hotChocolate.equals(coffee.getTemperature())

E hotChocolate.getTemperature().equals(coffee.getTemperature())

2、Consider the following class definition.

public class Document

{

private int pageCount;

private int chapterCount;

public Document(int p, int c)

{

pageCount = p;

chapterCount = c;

}

public String toString()

{

return pageCount + " " + chapterCount;

}

}

The following code segment, which is intended to print the page and chapter counts of a Document object, appears in a class other than Document.

Document d = new Document(245, 16);

System.out.println( /\* missing code \*/ );

Which of the following can be used as a replacement for /\* missing code \*/ so the code segment works as intended?

A d.toString()

B toString(d)

C d.pageCount + " " + d.chapterCount

D d.getPageCount() + " " + d.getChapterCount()

E Document.pageCount + " " + Document.chapterCount