11th Annual ViperWolf @ VHS

Invitational

March 7, 2020

General Directions:

1) DO NOT OPEN EXAM UNTIL TOLD TO DO SO.
2) **NO CALCULATORS of any kind may be used.**
3) You have 45 minutes to complete this contest. If you are in the process of actually writing an answer when the signal to stop is given, you may finish writing that answer.
4) Papers may not be turned in until forty-five minutes have elapsed. If you finish the test before the end of the allotted time, remain at your seat and retain your paper until told to do otherwise. You may use this time to check your answers.
5) All answers must be written on the answer sheet/Scantron card provided. Indicate your answers in the appropriate blanks provided on the answer sheet or on the Scantron card. Clean erasures are necessary for accurate Scantron grading.
6) You may place as many notations as you desire anywhere on the test paper except on the answer sheet or Scantron card which is reserved for answers only.
7) You may use additional scratch paper provided by the contest director.
8) All questions have ONE and only ONE correct (BEST) answer. There is a penalty for all incorrect answers. **All provided code segments are intended to be syntactically correct, unless otherwise stated (i.e. `error` is an answer choice). Ignore any typographical errors and assume any undefined variables are defined as used.**
9) A reference to commonly used Java classes is provided with the test and you may use this reference during the contest. You may detach the reference sheets from the test booklet but DO NOT DO SO UNTIL THE CONTEST BEGINS.
10) Assume that any necessary import statements for Standard Java 2 Packages and classes (e.g. .lang, .util, System, Math, Double, etc.) are included in any programs or code segments that refer to methods from these classes and/or packages.

Scoring:

1) All questions will receive 6 points if answered correctly; no points will be given or subtracted if unanswered; 2 points will be deducted for each incorrect answer.

This Page Intentionally Left Blank – Use as Scratch Paper

Standard Classes and Interfaces — Supplemental Reference

class java.lang.Object
- boolean equals(Object other)
- String toString()
- int hashCode()

interface java.lang.Comparable<T>
- int compareTo(T other)
  Return value < 0 if this is less than other.
  Return value = 0 if this is equal to other.
  Return value > 0 if this is greater than other.

class java.lang.Integer implements Comparable<Integer>
- Integer(int value)
- int intValue()
- boolean equals(Object obj)
- String toString()
- int compareTo(Integer anotherInteger)
- static int parseInt(String s)

class java.lang.Double implements Comparable<Double>
- Double(double value)
- double doubleValue()
- boolean equals(Object obj)
- String toString()
- int compareTo(Double anotherDouble)
- static double parseDouble(String s)

class java.lang.String implements Comparable<String>
- int compareTo(String anotherString)
- boolean equals(Object obj)
- int length()
- String substring(int begin, int end)
  Returns the substring starting at index begin and ending at index (end - 1).
- String substring(int begin)
  Returns substring(from, length()).
- 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 the specified index.. Returns -1 if str is not found.
- charAt(int index)
- int indexOf(int ch)
- int indexOf(int ch, int fromIndex)
- String toLowerCase()
- String toUpperCase()
- String[] split(String regex)
- boolean matches(String regex)

class java.lang.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 java.lang.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, in b)
- static int max(int a, int b)
- static long round(double a)
- static double random()
  Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.

interface java.util.List<E>
- boolean add(E e)
- int size()
- Iterator<E> iterator()
- ListIterator<E> listIterator()
- E get(int index)
- E set(int index, E e)
  Replaces the element at index with the object e.
- void add(int index, E e)
  Inserts the object e at position index, sliding elements at position index and higher to the right (adds 1 to their indices) and adjusts size.
- E remove(int index)
  Removes element from position index, sliding elements at position (index + 1) and higher to the left (subtracts 1 from their indices) and adjusts size.

class java.util.ArrayList<E> implements List<E>

class java.util.LinkedList<E> implements List<E>, Queue<E>

Methods in addition to the List methods:
- void addFirst(E e)
- void addLast(E e)
- E getFirst()
- E getLast()
- E removeFirst()
- E removeLast()

class java.util.Stack<E>
- boolean isEmpty()
- E peek()
- E pop()
- E push(E item)

interface java.util.Queue<E>
- boolean add(E e)
- boolean isEmpty()
- E peek()
- E remove()

class java.util.PriorityQueue<E>
- boolean add(E e)
- boolean isEmpty()
- E peek()
- E remove()

interface java.util.Set<E>
- boolean add(E e)
- boolean contains(Object obj)
- boolean remove(Object obj)
- int size()
- Iterator<E> iterator()
- boolean addAll(Collection<? extends E> c)
- boolean removeAll(Collection<?> c)
- boolean retainAll(Collection<?> c)

class java.util.HashSet<E> implements Set<E>

class java.util.TreeSet<E> implements Set<E>

interface java.util.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()

class java.util.HashMap<K,V> implements Map<K,V>

class java.util.TreeMap<K,V> implements Map<K,V>

interface java.util.Map.Entry<K,V>
- K getKey()
- V getValue()
- V setValue(V value)

interface java.util.Iterator<E>
- boolean hasNext()
- E next()
- void remove()

interface java.util.ListIterator<E> extends java.util.Iterator<E>

Methods in addition to the `Iterator` methods:
- void add(E e)
- void set(E e)

class java.lang.Exception
- Exception()
- Exception(String message)

class java.util.Scanner
- Scanner(InputStream source)
- boolean hasNext()
- boolean hasNextInt()
- boolean hasNextDouble()
- String next()
- int nextInt()
- double nextDouble()
- String nextLine()
- Scanner useDelimiter(String pattern)

A+ Computer Science Contest #1920-14 March 07, 2020

Note: Correct responses are based on Java, **J2sdk v 1.8.x**, from Sun Microsystems, Inc. All provided code segments are intended to be syntactically correct, unless otherwise stated (i. e. `error` is an answer choice) and any necessary Java 2 Standard Packages have been imported. Ignore any typographical errors and assume any undefined variables are defined as used. **For all output statements, assume that the System class has been statically imported…** ***import static java.lang.System.*;***

QUESTION 1

What is DE_{16} - AD_{16} ?

A. 33_{16} B. 1011010_2 C. 50_{10} D. 110001_2 E. 49_{16}

QUESTION 2

What is output by the code to the right?

A. 17.75 B. 18.0 C. 18 D. 17 E. 17.0

```
double x;
x = 5 + 7 % 9 * 9 / 4 - 3;
out.println(x);
```

QUESTION 3

Which of the following correctly replaces <*1> in the code to the right such that the output is

```
01234567890123456789
GAVE          WARNING
 THEYFAIR
```

A. `out.printf("5s-15s\n-5s15s",b,a,c,d);`

B. `out.printf("%5s%-15s\n%-5s%15s",c,b,a,d);`

C. `out.printf("-5s15s\n5s-15s",c,b,a,d);`

D. `out.printf("%-5s%15s\n%5s%-15s",b,d,a,c);`

E. `out.printf("%-5s%15s\n%5s%-15s",b,a,c,d);`

```
String x = "01234567890123456789";
String a = "THEY";
String b = "GAVE";
String c = "FAIR";
String d = "WARNING";
out.println(x);
<*1>
```

QUESTION 4

What is output by the code to the right?

A. RACKETMON

B. RACKETMAN

C. ROCKETMAN

D. RCKETMN

E. There is no output due to a compile time error

```
String x = "ROCKETMAN";
out.println(x.replace('O', 'A'));
```

QUESTION 5

What could replace the line of code at <*1> in the code to the right?

A. `out.println(false);`

B. `out.println(a || b);`

C. `out.println(b);`

D. `out.pritnln(true);`

E. `out.println(!a || b);`

```
boolean a, b;
a = ???;
b = ???;
out.println(a || !a && b); //<*1>
```

QUESTION 6

What is output by the code to the right?

A. 24 B. -23.0 C. -24.0 D. 23 E. -30

```
out.println(Math.floor(-23.1039));
```

QUESTION 7

What is output by the code to the right?

A. 39 B. 41 C. 43 D. 42

E. there is no output due to a compile time error

```
int x = 39;
double y = 2.9;
x += y;
out.println(x);
```

QUESTION 8

What could be the values of x and y such that the code to the right outputs VADER ?

A. x = "Yoda"; y = "yoGurt";

B. x = "yoGurt"; y = "yodA";

C. x = "yoda"; y = "yogurt";

D. x = "yogurt"; y = "yoda";

E. x = "Yogurt"; y = "yoda";

```
String x, y;
x = "yogurt"; y = "yoda";
if(x.compareTo(y)<0)
out.println("YODA");
else if(x.compareTo(y)>0)
  out.println("VADER");
else
  out.println("LONE STAR");
```

QUESTION 9

How many times is BOB printed out by the code to the right?

A. 7 B. 8 C. 9 D. 10 E. 11

```
for(int i=0; i<25; i+=3)
  out.println("BOB");
```

QUESTION 10

What is output by the code to the right?

A. 3 B. 6 C. 1 D. 4 E. 5

```
int[] list = {4,8,2,5,3,3,4,5,0,5,1};
list[6] = list[list[7]]+1;
out.println(list[6]);
```

QUESTION 11

What could replace the <1> in the code to the right such that it will read from a text file called empire?

A. new Scanner("empire.txt")

B. new File(new Scanner("empire.txt"))

C. new Scanner(new File("empire.txt"))

D. new File("empire.txt")

E. more than one of these will work

```
Scanner input = <1>;
```

QUESTION 12

What is output by the code to the right?

A. 19535040

B. 27907200

C. 1395360

D. 0

E. 390700800

```
int x = 0;
for(int i=15; i<20; i++)
  x*=i;
out.println(x);
```

QUESTION 13

Which of the following operators does not have the same level of precedence as the rest of the operators?

A. ~ B. ++ C. -- D. >> E. !

QUESTION 14

What is output by the code to the right?

A. `44.0`

B. there is no output, because an `int` primitive data type can't autobox to a `Double` data type

C. there is no output because the `Double` data type needs to be instantiated

D. there is no output because the `Integer k` cannot be unboxed into an `int` primitive data type

E. there is no output because an `Integer` data type needs to be cast into a `Double` data type.

```
Integer k = 96;
Double z;
z = k - 52;
out.println(z);
```

QUESTION 15

What is output by the code to the right?

A. `10.6`

B. `7.1`

C. `38.5`

D. `19.4`

E. There is no output due to a run time error

```
ArrayList<Double> x =
            new ArrayList<Double>();
x.add(7.1); x.add(10.6);  x.add(4.5);
x.add(2.3); x.add(38.5); x.add(19.4);
out.println(x.remove(1));
```

QUESTION 16

What is output by the code to the right?

A. `51` B. `17` C. `32` D. `53` E. `34`

```
out.println(34&53);
```

QUESTION 17

How many combinations of (A,B,C) result in the value true for the following Boolean expression?

```
(A && B || !(A || C)) && !B
```

A. `5` B. `0` C. `6` D. `1` E. `8`

QUESTION 18

What is output by the code to the right?

A. `[M, P, R]`

B. `[P, R]`

C. `[E,M,P,I,R,E]`

D. `[E, I, E]`

E. There is no output due to a runtime error

```
String[] x;
x = "EMPIRE".split("");
ArrayList<String> list =
                    new ArrayList<>();
for(String k:x)
  list.add(k);
list.remove(0);
for(int i=0; i<list.size(); i++)
  if(list.get(i).compareTo("L")<0)
     list.remove(i);
out.print(list);
```

QUESTION 19

What could replace the `<*1>` in the code to the right such that it will print out `12:0:3`?

A. `c = m/6; c %= 6; p = m/3; c %= 3;`

B. `m /= 6;  c %= 6; c /= 3;  p %= 3;`

C. `c = m/6; m %= 6; p = c/3; c %= 3;`

D. `p = m/6; m %= 6; c = m/3; c %= 3;`

E. `p = m/6; c %= 6; m = c/3; c = m % 3;`

```
// ratios
//6 m : 1 c
//3 c : 1 p

int p, c, m;
m = 219;
// <*1>
out.println(p+":"+c+":"+m);
```

QUESTION 20

What is output by the code to the right?

A. `3428134283423430`

B. `342813428342343`

C. `342812`

D. an infinite loop

E. There is no output due to a compile time error

```
int x = 342812;
while(x!=0)
  out.print(x/=10);
```

QUESTION 21

What is output by the code to the right at line `//<*1>`?

A. `756828076`

B. `NAOMI`

C. `QXLJF`

D. `KDRPL`

E. There is no output due to a run time error

QUESTION 22

What is output by the code to the right at line `//<*2>`?

A. `KHFRPH`

B. `enlxvn`

C. `69786888678`

D. `ENLXVN`

E. `khfrph`

```
public String mys1(String x, int num)
{
  int val = x.charAt(0)-'A';
  if(val % 2 == 0)
     val = (val+num)%26;
  else
     val = (val-num+26)%26;
  char k = (char)('A'+val);
  return ""+k;
}

// CLIENT CODE
String str = "NAOMI";
String str2="";
for(int i=0; i<str.length(); i++)
 str2+=mys1(str.substring(i,i+1), 3);
out.println(str2); //<*1>

str = "become";
str2="";
for(int i=0; i<str.length(); i++)
 str2+=mys1(str.substring(i,i+1), 3);
out.println(str2); //<*2>
```

QUESTION 23

Which of the following lines of code will compile based on the code to the right?

```
I.  L x = new A();
II. C x = new A();
III.Y x = new A();
IV. S x = new A();
V.  A x = new A();
```

A. `III, IV, V`

B. `II, III, IV, V`

C. `I, II, III, IV, V`

D. `II, V`

E. `V only`

```
class A extends C
{
}

class C
{
}

class L extends C
{
}

class S extends A
{
}

class Y extends A
{
}
```

QUESTION 24

The programmer of the code to the right has created an abstract class called `T` and wants only S and Y to additionally inherit that class. What should the programmer do to achieve this result?

A. have `T extend S` and `T extend Y`

B. have `S extend A, T` and `Y extend A, T`

C. have `C extend T`

D. have `A extend C, T`

E. It cannot be done because `S` and `Y` can only inherit one class

QUESTION 25

What is output by the code to the right?

A. `[34, 28, 6, 47]`

B. `[34, 28, 47]`

C. `[34, 28, 0, 0, 47, 0]`

D. a memory location

E. There is no output due to a compile time error

```
int[] list = {34, 28, 12, 6, 47, 24};
for(int i=0; i<list.length; i++)
  if(list[i]%6 == 0)
     list.remove(i);
out.println(list);
```

QUESTION 26

What is output by the code to the right?

A. `342813428134281`

B. `318243182431824`

C. `20191817161514131211109876`

D. `182431824318243`

E. There is no output due to an out of bounds error

```
int[] list = {3, 4, 2, 8, 1};
int x = 20;
while(x>5)
{
  out.print(list[x%list.length]);
  x--;
}
```

QUESTION 27

Which of the the following line of code would print `0` when calling the code to the right?

A. `out.println(mys2(1,5));`

B. `out.println(mys2(2,3));`

C. `out.println(mys2(2,4));`

D. `out.println(mys2(1,4));`

E. `out.println(mys2(3,3));`

```
public static int mys2(int x, int y)
{
  if(x>=y)
     return 0;
  return x*y+mys2(x+1,y)+mys2(x,y-1);
}
```

QUESTION 28

Which of the the following line of code would print `129` when calling the code to the right?

A. `out.println(mys2(1,5));`

B. `out.println(mys2(2,3));`

C. `out.println(mys2(2,4));`

D. `out.println(mys2(1,4));`

E. `out.println(mys2(3,3));`

QUESTION 29

What is output by the code to the right at line `//<*1>`?

A. `30` B. `28` C. `38` D. `41`

E. There is no output due to a run time error

QUESTION 30

What is output by the code to the right just before line `//<*2>`?

A. `30 38 39 41`

B. `41 5 22 38`

C. `5 22 38`

D. `39 41`

E. There is no output due to a run time error

```
Set<Integer> set = new TreeSet<>();
set.add(30); set.add(10); set.add(39);
set.add(7); set.add(28);
set.add(41);set.add(5); set.add(22);
set.add(38);
Iterator<Integer> i = set.iterator();
i.next(); i.next(); i.remove();
i.next(); i.next(); i.remove();
out.println(i.next()); //<*1>
i.forEachRemaining(k->out.print(k+" "));
//<*2>
```

QUESTION 31

What is output by the code to the right at line //<*1>?

A. MR

B. [AR, LT, AC, AT, MB, MR]

C. TA

D. [AC, AR, AT, LT, MB, MR]

E. There is no output due to a compile time error

QUESTION 32

What is output by the code to the right at line //<*2>?

A. [HA, LA, SM, PE, NG]

B. [HA, LA, NG, PE, SM]

C. [PE, NG]

D. [NG, PE]

E. There is no output due to a compile time error

```
Map<String,TreeSet<String>> m;
m = new TreeMap<String,TreeSet<String>>();

m.put("T", new TreeSet<String>());
m.put("D", new TreeSet<String>());
m.put("TA", new TreeSet<String>());

TreeSet<String> x = m.get("T");
x.add("HA"); x.add("LA"); x.add("SM");
x = m.get("D");
x.add("PE"); x.add("NG");
x = m.get("TA");
x.add("AR"); x.add("LT"); x.add("AC");
x.add("AT"); x.add("MB"); x.add("MR");
out.println(m.get("TA")); //<*1>

m.put("DC",m.get("T").addAll(m.get("D")));
out.println(m.get("D")); //<*2>
```

QUESTION 33

What is output by the code to the right?

A. 28

B. 38

C. 23

D. 30

E. 10

```
Queue<Integer> queue;
queue = new LinkedList<>();
queue.add(10); queue.add(30);
queue.remove(); queue.add(28);
queue.add(queue.peek());
queue.add(23); queue.add(28);
queue.add(38); queue.remove();
queue.peek(); queue.remove();
out.println(queue.peek());
```

QUESTION 34

What is output by the code to the right?

A. 1

B. 0

C. 3

D. 5

E. 2147483647

```
out.println(Integer.MAX_VALUE>>29);
```

QUESTION 35

How many empty Strings are created and placed into list by the code to the right?

A. 4 B. 0 C. 5 D. 9 E. 2

```
String x = "BANANAALABAMAATLANTA";
String [] list;
list = x.split("[BA]|[LA]");
```

QUESTION 36

Simplify the Boolean algebra statement to the right.

A. !A

B. true

C. !A!BC + B!C

D. !AB^C

E. false

```
!(A+B)(!B(A+C))+!CB
```

QUESTION 37

Solve the following expression?

```
LCIRC-2(201 XOR 221) OR RSHIFT-2(236)
```

A. 80 B. 123 C. 236 D. 197 E. 222

QUESTION 38

What is the signed byte binary representation of -101?

A. 10011011

B. 01101010

C. 01111011

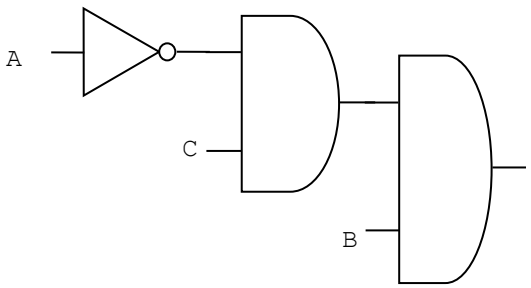
D. 10011001

E. 11100101

QUESTION 39

OPEN ENDED QUESTION – Find the answer and write it on your answer sheet. If you are using a ScanTron form, write the question number and the answer on the bottom of the ScanTron.

Write all the ordered triplets that will make the digital circuit diagram to the right true. Write a triplet in the form of (A, B, C), where A, B, and C will either be a 0 or a 1. For example: (1,1,1) or (0,1,1).



QUESTION 40

OPEN ENDED QUESTION – Find the answer and write it on your answer sheet. If you are using a ScanTron form, write the question number and the answer on the bottom of the ScanTron

Solve the equation to the right which is written in prefix notation.

```
- / + 21 19 10 * 5 7
```