

Computer Science Contest #1314-01 Key

October 12, 2013

- |       |       |
|-------|-------|
| 1) C  | 21) B |
| 2) D  | 22) A |
| 3) C  | 23) D |
| 4) C  | 24) C |
| 5) B  | 25) A |
| 6) E  | 26) A |
| 7) D  | 27) D |
| 8) D  | 28) A |
| 9) C  | 29) D |
| 10) E | 30) A |
| ■     | ■     |
| 11) A | 31) B |
| 12) B | 32) E |
| 13) C | 33) E |
| 14) D | 34) C |
| 15) C | 35) D |
| 16) E | 36) D |
| 17) C | 37) E |
| 18) C | 38) A |
| 19) C | 39) C |
| 20) D | 40) E |
| ■     | ■     |

**Note to Graders:**

- All provided code segments are intended to be syntactically correct, unless otherwise stated (e.g. error is an answer). **Ignore any typographical errors.**
- Any necessary Standard Java 2 Packages are assumed to have been imported as needed.
- Assume any undefined (undeclared) variables have been defined as used.

Brief Explanations:

1.  $132_{10} = 10000100$ ,  $10000100 + 1111_2 = 10010011_2 = 147_{10} = 93_{16}$
2. Java 7 lets you enter a binary number with the prefix 0b, however it will printout the normal decimal number
3.  $29/4$  is 7, then  $13\%7$  is 6 (the remainder)
4.  $k = 35, 27, 19, 11, 3, -5$ . At -5, it escapes the loop.
5. substring uses the start index all the way up to, but exclusive of, the stopping index.
6.  $n[1] = 5$ ,  $n[5] = n[8]$ ,  $n[8]$  is out of bounds which is a run time error
7.  $x \ \&\& \ z \ || \ z \ \&\& \ (!x \ || \ x \ \&\& \ y) = xz + z(!x + xy) = xz + !xz + xyz = z(x + !x + xy) = z(1 + xyz) = z(1) = z$ . Boolean algebra is your friend.
8. switch will look up the value in the cases and do the assign statement.
9. The ? is a short hand for if, the : is an else. It is similar to saying `if(x%2==0) str+="A"; else str+="B";`
10. hits is a private variable and cannot be called through the object, so you must calculate it with the available methods.
11. the value is between [13,55) because 41.999... is the highest value random can make and 13 is the amount the number is shifted to the right.
12. just because the last k is not used, does not cause this code to error.
13. The character 5 is the integer value 53,  $53\%5$  is 3 which is concatenated onto the String ATE
14. k counts up starting at location (3,1), it continues to fill up the first column and then moves to the next column (3,2).
15. Every time  $x \geq y$ , 0 will be returned. Do not assume the first number sent up will be the first number used in the for loop.
16. finally will always occur, no matter if try or catch work.
17. 35 is 100011 and 15 is 001111, the xor is 101100 which is 44.
18.  $(a+b)!c+c(!c+a+b) \rightarrow a!c+b!c+c!c+ac+bc \rightarrow a!c+b!c+1+ac+bc \rightarrow a(!c+c)+b(!c+c) \rightarrow a(1)+b(1) \rightarrow a+b$ . So, in order to be true a or b must be true. If neither of them are true, then d is false.
19. `x.add(x.remove(5))` will remove 0 from the array and then add it to the end.
20. % is modulus and returns the remainder value of the long division.
21. The loop would check o,e,s,' ',i, and then t. Which would create 5.
- 22 & 23. The trick to this problem is to know that the value of 'A' is 65 and the value of 'a' is 97. From there you can figure out all the values.
24. You must call `super.getMana()`, if you call `getMana()` you will be in an infinite loop.
25. instanceof is used to check if an object is of a specific class.
26. The loop keeps adding up the energy. Once energy is used up, it is removed from the ArrayList.
27. The first loop sets up an array that counts up the number of letters, with each index representing a letter in the alphabet-1.
28. The second loop is going through the array and adding the non-zero letters to an ArrayList.
- 29 & 30. This recursion will continue to cut the string in half until it hits a 5 or is of size one or less. However, it also skips a character when splitting the string (num-1). If it hits a five, it returns the value 2, if it is size one or less it returns a 0. Add them up to get the result.
31. We are working with a size 4 times the original size. The method is quadratic, so we must square the size increase. So the time will be 16 times the speed of the original time.  $16 \times 5$  is 80 seconds.
32. A pre-order traversal prints out the value, then visits the left child, then the right child. So the parent node is printed before the braches are visted.
33. The add method works just like the push method on the Stack, which means the bottom of the stack is at index 0.

- 34. The pop method removes from the end of the list. The push method adds to the end of the list.
- 35. peek only looks at the element, and does not remove it.
- 36. A PriorityQueue is a min heap, which means it is held as a balanced binary tree, with each parent smaller than its children.
- 37. In a PriorityQueue, you always remove the smallest value. In this case we are removing the third smallest value which is abcd.
- 38.  $x \gg y$  is like saying  $x / 2^y$ . So it is  $40892/16=2555$ . And only one answer is even remotely possible.
- 39&40. The array splits on every 'l', except the last two 'l's, where it does not make an extra empty String. [we, cometota, "", yha]. That is an important thing to know about split.