**CS Technical Interview Questions**

1. What is an object?
   1. An object is code that combines data, state, and behavior.
   2. It is an instance of a class, created by calling a method within that class.
2. What will this code do?

for (int i = 0; 9; i++) {

…

}

a) This code will not compile because the conditional statement within the for-loop declaration does not return a Boolean value.

1. Components of a for-loop?
   1. Keyword “for”, initialization expression, termination expression, increment expression
2. What is a constructor?
   1. A special method used to initialize the state of objects of a class.
3. What will be the output of this code?

int wow (int x, int y) {

return x \* y;

}

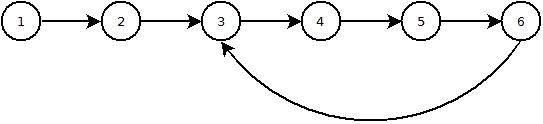
byte a = 10;

byte b = 20;

System.out.println(wow(a, b));

* 1. 200
  2. Java supports widening primitive conversion, meaning a method passing in a specified data type can also accept its subset primitive data types (e.g., bytes are subsets of integers).

1. What is a static variable?
   1. A variable that belongs to the class, which is shared by all instances of the class.
2. What does the keyword “this” do?
   1. It refers to the current object of a method/constructor.
3. What is the difference between a linear and non-linear data structure?
   1. Linear data structures are arranged sequentially, whereas non-linear data structures are not.
4. Are linked lists linear or non-linear data structures?
   1. Non-linear
5. What is difference between a stack and a queue?
   1. A stack has the same spot for insertion and removal while a queue has different insertion and removal spots.
6. How can you tell if a linked list has a loop?



* 1. Have two pointers starting at the head of the linked list. For every incrementation, one pointer traverses one node while the other traverses two nodes. If at any point both pointers point to the same node, there is a loop.

1. How can you find the nth element in a linked list from the end with only one traversal?
   1. Have two pointers starting at opposite ends of the linked list. Traverse both pointers by one node towards their respective opposite ends until they point to the same node.