

CSE143X Lecture Questions

Time (e.g., 12:45)	Question	Answer
9:50	Why does && take precedence over ? Why are they not on the same level of precedence? Got it, thanks!	We generally like to have different levels of precedence for different operators. Think of and && as being like + and * for logic.
21:30	In obj #, how does java decide the #? Ok thank you	Java does not give numbers to objects. This is a jGRASP feature. You could think of it as an indication of where it is stored in memory.
20:00	Is there any advantage to making Strings immutable? Makes sense thank you	Absolutely. Suppose I have a company record that has my name and mailing address as strings. If you are allowed to access them and change them, then you could redirect my paychecks to go to you instead of me. It's good to know that strings are protected and can't be altered, so you are always willing to give someone access to a string.
15:20	In your example, indexOf(str) get the index of a character. So if the input str has more than one character? thanks.	It searches for that complete string and returns the index of the first occurrence if there is one (starting index).
	What was the Java equivalent of Python's IDLE called again? Thanks	In jGRASP it's the interactions pane. For standard Java it's jShell.
	Is there an easy way to return multiple values (i.e. two ints) from a method? (that we are allowed to use)?	No. You can return only one thing from a Java method. Java's solution is to package multiple data items into a single object or put them in a structure like an array, but we won't get to that until we hit chapters 7 and 8.
	^ arrays get introduced next Wednesday, before the next assignment is due. Can we use them for this purpose?	No (homework 3 goes through chapter 5)

	<p>Are we allowed to use break / continue since we have while loops now? Okay, that's fine, but can I ask why? Ok</p>	<p>No. They are on the list of forbidden constructs under "general style deductions". As that web page mentions, we want to have all students using the same constructs (level playing field) and we don't want to cover everything.</p>
	<p>can we use something like $x < (a \parallel b)$ ok, thank you!</p>	<p>No. The \parallel operator requires two boolean expressions, so you would need $(x < a) \parallel (x < b)$</p>
	<p><code>S = s.toUpperCase()</code> let s refer to a new string "HELLO THERE", but the old string "hello there" is still there, i think it's redundant. Why not just change the string itself, but to create a new one? gotcha</p>	<p>It is often the case that a Java program loses the reference to some object. There is something known as the garbage collector that reclaims unused objects like these. Having immutable strings takes up more memory, but has its own advantages (see above). A Java guru named Joshua Bloch has as one of his top tips for Java that you should "favor immutability."</p>
48:37	<p>Can you briefly explain again why the simulator acts that way--as if it was the casino itself playing and always getting money when the program isn't edited to not include 0? Maybe I am just unfamiliar with the way casinos really work, but I thought 0 just serves as a digit and let us say if you were to have 37 digits from 1-37, spinning a 1 wouldn't make you infinitely gain money? Re: riighttt ok gotchu! Brb telling my brother not to go anymore.</p>	<p>The Roulette wheel has 37 numbers on it. When you bet low, you're betting that it's between 1 and 18. That's 18 numbers, which gives you a probability of 18/37 of winning. That's slightly below 50% chance. If you bet that over and over, eventually you'll lose. By including 0 as a win, we had 19 numbers that were winners. That made the probability 19/37, which is slightly higher than 50%. That's how casinos make money. The odds are tipped slightly in their favor and over time they end up winning a lot of money.</p>
34:25	<p>Can I write <code>s = s.substring + replacement + substring; return s</code> and in main function just call replace function?</p>	<p>I'm not exactly sure what you're asking. You can reset s in the method and return it as an alternative to what we did, but that will not change s in main.</p>