All quiz rules from the course syllabus are in effect for the real quiz, in addition to what follows.

If you have questions about the test, please ask *on Piazza* as a **private** post, viewable only by you and the instructors. The instructors will make an announcement via a pinned errata post if something needs to be said to the entire class. **Do not make comments on any non-private posts on Piazza during the quiz for any reason. Making such comments may subject you to a grade penalty.**

This quiz is to be individual effort. Students are permitted to use notes, electronics, and bring textbooks. The work you submit for each quiz is expected to be produced by *you, alone and solely for this assessment*. You may not reuse or repurpose anything you wrote at another time, nor may you ever use anything written by someone else during this quiz. However, despite being allowed notes and electronics, you may *not* seek out the answer to a question in any way, nor may you communicate with anyone during the exam, *for any reason*, with the exception of asking a question on Piazza *set as instructors-only for visibility*.

if you submit anything for free response that you did not produce during this exam, you must cite the source of any aspect(s) of it very clearly; failure to do so constitutes academic dishonesty.

You will have 40 minutes for this quiz, plus an additional ten minutes to enter your answers to the answer form. Please be careful when you enter your answers, as this will very likely be graded by a computer program and not a human. Students who have timed accommodations, such as through DSC, should use the 50 minute total as a baseline and apply their accommodations from there.

Submit your answers to this Google Form:

`https://docs.google.com/forms/d/e/1FAIpQLScBqkn1e2jziw0ZDIygZDyu584zNEW0tuMVGQJS33fR6DUQdQ/viewform`

You may need to log in with your UCI credentials to access this. You may re-submit if you like, but your most recent submission is the one that will be graded. This means if you submit or re-submit after the end of the quiz, you will have a zero.

When 30 minutes have passed from the start of the quiz, check Piazza to see if there is a pinned thread that has clarifications to the questions. Anything on that thread by that time is something you are responsible for.

Constrained Answer Questions

Fill in your answers here on the Google Form referenced on the cover page.

1. (1.5 p**o**ints) Suppose we are computing the LCS (Longest Common Subsequence) of tw**o** strings, using the algorithm from class. We find the f**o**ll**o**wing table, alth**o**ugh I have **o**mitted the letters from the word that would be printed acr**o**ss the top.

		X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
	0	0	0	0	0	0	0	0	0	0
h	0	1	1	1	1	1	1	1	1	1
e	0	1	1	1	1	1	1	1	2	2
r	0	1	1	2	2	2	2	2	2	3
b	0	1	1	2	3	3	3	3	3	3
i	0	1	1	2	3	4	4	4	4	4
v	0	1	1	2	3	4	4	4	4	4
o	0	1	1	2	3	4	4	4	4	4
r	0	1	1	2	3	4	4	4	4	5
e	0	1	1	2	3	4	4	4	5	5

   (a) Identify a letter fr**o**m "herbivore" other than 'r' that appears in X

   (b) Identify a letter from "herbivore" th**a**t **does not** appear in X.

   (c) Which index of X c**o**rresp**o**nds t**o** the letter 'r'? If it appears multiple times, select only one.

2. (1.5 points) Suppose I am computing edit distance using the algorithm from lecture. However, this time I have omitted both strings from the chart and have instead written the string via indices.

   Recall that the iterative code is as follows.

   EditDistance($X_{1\ldots n}$, $Y_{1\ldots m}$)
   for $j \leftarrow 0 \ldots m$ **do**
   Edit[0, j] $\leftarrow j$
   for $i \leftarrow 1 \ldots n$ **do**
   Edit[i, 0] $\leftarrow i$
   for $j \leftarrow 1 \ldots m$ **do**
   ins $\leftarrow$ 1+ Edit[$i, j-1$]
   del $\leftarrow$ 1+ Edit[$i-1, j$]
   sub $\leftarrow$ Edit[$i-1, j-1$] + ($X_i \neq Y_j$)
   Edit[i, j] $\leftarrow$ min(ins, del, sub)

   For each question, answer "yes" if the answer is definitely "yes," "no" if it is definitely "no" and "maybe" if there isn't enough information to answer definitively. The questions are all asking if the elements asked about are the same letter in both words. Both X and Y are English words.

		X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
	0	1	2	3	4	5	6	7	8	9	10
Y_1	1	0	1	2	3	4	5	6	7	8	9
Y_2	2	1	1	2	3	4	5	5	6	7	8
Y_3	3	2	2	2	3	3	4	5	6	7	7
Y_4	4	3	3	3	3	4	4	5	6	7	8
Y_5	5	4	4	4	4	4	4	5	6	7	8
Y_6	6	5	5	5	5	5	4	5	6	7	8
Y_7	7	6	6	6	6	6	5	4	5	6	7
Y_8	8	7	7	7	7	7	6	5	5	6	7
Y_9	9	8	7	8	8	8	7	6	5	6	7
Y_{10}	10	9	8	8	9	9	8	7	6	6	7

   (a) $X_2 == Y_2$?

   (b) $X_6 == Y_5$?

   (c) $X_9 == Y_{10}$?

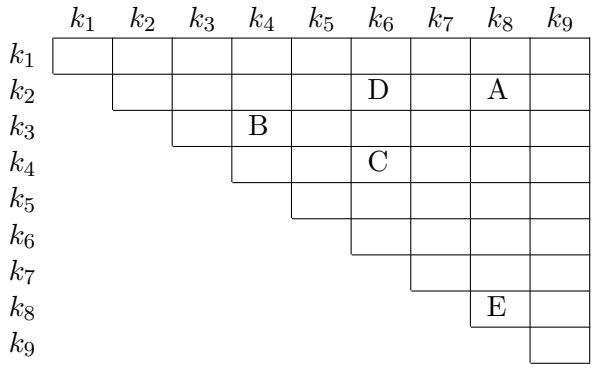
3. (1 point) Consider the following table, which is the output from running the optimal binary search tree algorithm for some input. In each entry, the value printed in the upper half of the cell is OPT$[i, j]$, the cost of the optimal binary search tree consisting of keys $i \ldots j$. The value printed in the lower half of the cell is the value of roots$[i, j]$.

   Suppose we have computed the following using the algorithm for the optimal binary search tree. However, we accidentally forgot to record OPT$[1, n]$. What is the missing value for that spot? It is the only one omitted in the following table. On the form, write only the number.

   Optional Additional: If you are worried you may have miscalculated, but you think you got the work right, there is a second free-response portion. Write using ONLY numbers (including decimal points) and the + sign if you are using that. I will **attempt** to give appropriate partial credit with this (initially with my grading program, then by hand if we need to). You may leave this blank at no penalty. I am still figuring out how to use this.

	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8
k_1	0.12 1	0.37 2	0.67 2	1.07 3	1.49 3	1.83 3	2.31 4	 4
k_2		0.13 2	0.41 3	0.71 3	1.12 4	1.45 4	1.91 5	2.06 5
k_3			0.15 3	0.45 3	0.73 4	1.06 4	1.52 5	1.67 5
k_4				0.15 4	0.43 4	0.66 5	1.07 5	1.22 5
k_5					0.14 5	0.36 5	0.69 6	0.84 6
k_6						0.11 6	0.37 7	0.47 7
k_7							0.15 7	0.25 7
k_8								0.05 8

4. (1 point) Suppose **I** am going to run the Optimal BST algorithm for nine keys. Some **o**f the entries in this table are labeled with lett**e**rs. In which order will the iterative algorithm compute a value for these**,** filling in the entry**?** Write only the letters in the form in the order the algorithm will compute their values.



Free Response

Did you ever wonder if your CompSci 161 professor considered a career as a professional wrestler? Well, wonder no more: he didn't. But if he had, the knowledge from CompSci 161 might have helped him, as the following problem illustrates.

Suppose I am an aspiring professional wrestler and I am invited to be an extra in a large production. The total performance, including scenes I am not in, will be a total of n acts (for some positive integer n). Each act needs one extra to perform some stunt, and I am allowed to choose which scene(s) I wish to perform in. For each of the n acts, I will be paid $\$m_i$ if I participate (and \$0 if I do not). However, each act takes some $p_i > 0$ (it is also the case that $p_i < i$) time to prepare, measured in how many acts prior to this one I cannot participate in if I wish to help in this one. For example, if $p_9 = 4$, if I perform the stunt in act nine, I cannot perform in acts five through eight. There is not necessarily a relationship between preparation time and money: for example, a clothesline might pay well but not require time to set up, while a splash off the top rope might take longer to set up but not pay as well.

If you want to picture this as a C++ program, the input might look like a `std::vector<Act>`, where `Act` is:

```
struct Act{
   unsigned p; // number of acts needed to prepare this.
   double m; // I will be paid $m if I do this act.
};
```

and n is the size of the `std::vector`.

Hint: should I perform the stunt in the final (nth) act?

Give a **dynamic programming** algorithm that takes as **input** the list of n stunts I am going to be asked to do, as well as the pay and setup time for each, and returns the optimal dollars I can earn by doing so. You do **not** need to output the set of stunts.

Please be sure to do the following for your response: (1) write out the base case and recurrence expressions; (2) give an English specification of the underlying recursive sub-problems and (3) analyze the running time for the iterative version, explaining any important implementation details. You do not need to actually write out the iterative version. However, you might find it useful to do so.

Remember to upload your response to GradeScope and tag your answer by the end of the quiz. The quiz is due at that time, not at the last time that GradeScope may or may not allow an upload. If you do not tag your response, or you do not upload one, your score for this question will be zero, regardless of other considerations.

If your entire response to this question is "I do not know," you will get 2 points out of 5 instead. Note that it is possible to earn fewer than 2 points for a bad response to this question. In order to get these points, your entire response must be "I do not know" – you may not say this and also provide an answer.