Effective Learning

Predict the outcome

Consider programming, as taught in many schools today. The teacher explains the syntax and the logic of a program, then asks students to go through the solved examples in the book, and gives a few exercises from the end of the chapter as homework.

In the exam, the students are required to write a program for one of the textbook problems.

Are these students likely to score well in the exam?

1. Yes
2. No

Predict the outcome

The same students are now given a problem that is unfamiliar to them but at the same level as the textbook problems.
They have to write a program, run it for a few test cases and give the output.

Are these students likely to be successful?
1. Yes
2. No

Discussions

Where are the disconnects?

Discussions

Your colleague says: I explained my topic well. I gave examples, solved problems, and asked questions. The responses showed that students understood. Then I was shocked to find that they have done miserably in the exams.

What could be some reasons for this?

Some terminology

1) What should they be learning?

- Can we define it precisely?

Course outcomes/ Learning outcomes

2) What can we do to help them learn it, as defined above?

Instructional strategies

3) Are they learning it?

- How do we know that they have learned it, as defined above?

Assessment

Some terminology

1) What should they be learning?

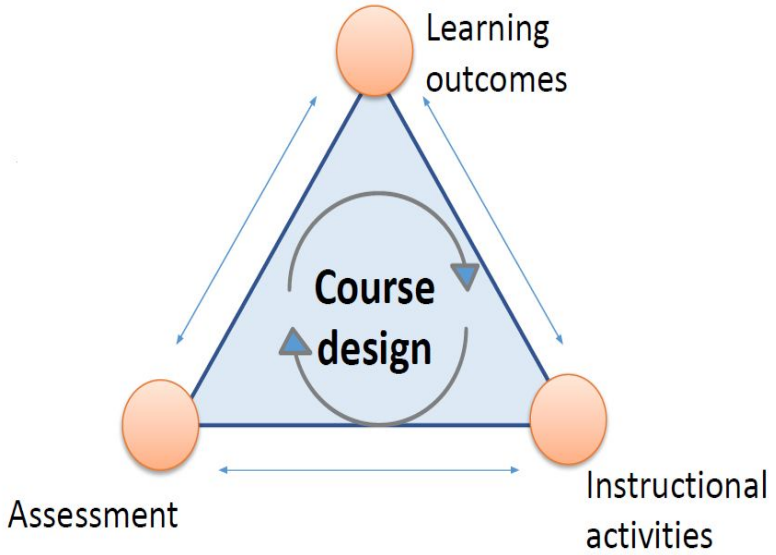
- Can we define it precisely?

2) What can we do to help them learn it, as defined above?

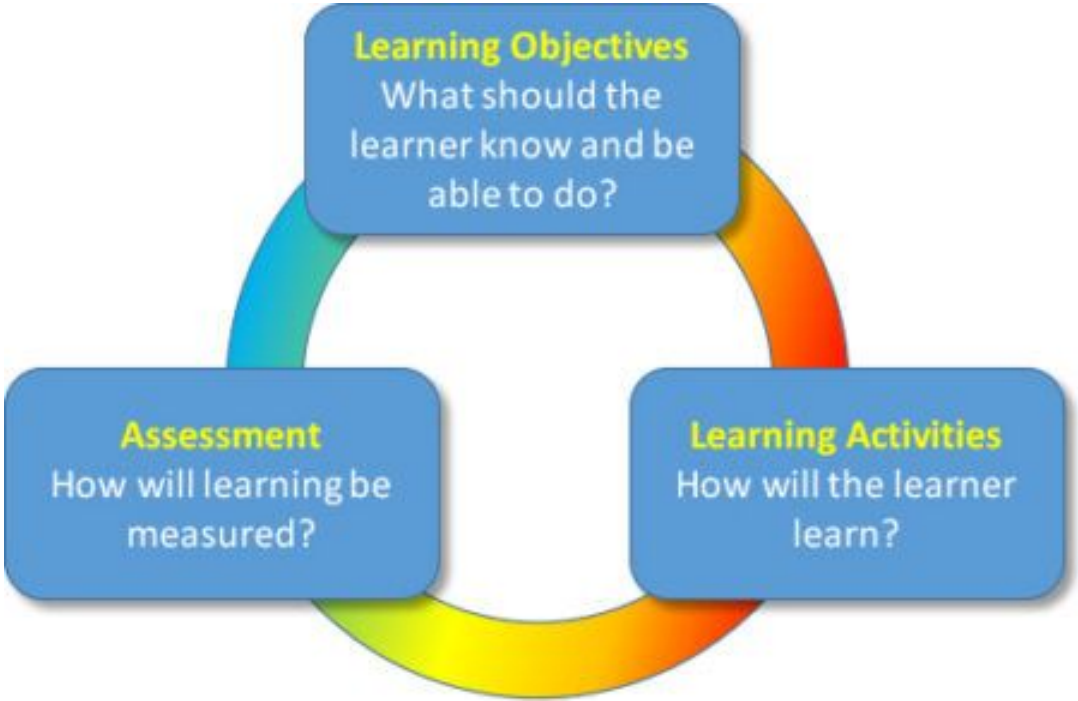
3) Are they learning it?

- How do we know that they have learned it, as defined above?

Triangle of effective learning



Constructive Alignment



The importance of constructive alignment

Constructive alignment is important for several reasons:

- **Enhanced Learning:** When learning outcomes, teaching activities, and assessment methods are aligned, students are **more likely to engage in deep learning** and develop a **thorough understanding** of the subject matter. They understand what is expected of them and how their learning will be assessed.
- **Improved Student Motivation:** When students see a clear connection between what they are learning, how they are being taught, and how they are being assessed, they are more likely to be **motivated to learn**. They understand the **relevance of the material** and are more likely to invest their time and effort in the learning process.
- **Fair and Valid Assessment:** When assessment methods are aligned with learning outcomes, they provide a fair and valid measure of student learning. Students are assessed on what they have been taught and given opportunities to **demonstrate their knowledge, skills, and abilities in meaningful ways.**
- **Effective Teaching:** Constructive alignment provides instructors with a framework for designing effective learning experiences. By carefully considering the learning outcomes, teaching activities, and assessment methods, instructors can create a **cohesive and impactful learning environment** that **fosters student success.**

Examples of constructive alignment

- **Learning Outcome:** Students will be able to design a website using HTML and CSS.
- **Teaching Activity:** Students attend lectures and hands-on workshops on HTML and CSS, and practice coding exercises.
- **Assessment Method:** Students create a functional website that meets specific design requirements, demonstrating their understanding of HTML and CSS.

Is there a constructive alignment?

- **Learning Outcome:** Students will be able to design a website using HTML and CSS.
- **Teaching Activity:** Students attend lectures and ppts for HTML and CSS, and fixed set of lab exercises.
- **Assessment Method:** Students create a functional website that meets specific design requirements, demonstrating their understanding of HTML and CSS.

Answer:

1. **Yes**
2. **No**

Is there a constructive alignment?

- **Learning Outcome:** Students will be able to design a website using HTML and CSS.
- **Teaching Activity:** Students attend lectures and ppts for HTML and CSS, and fixed set of lab exercises.
- **Assessment Method:** Students create a functional website that meets specific design requirements, demonstrating their understanding of HTML and CSS.

Answer:

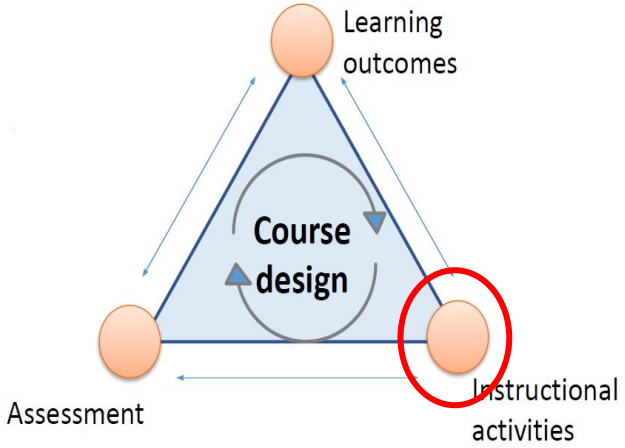
1. **Yes**
2. **No** **WHY?**

Is there a constructive alignment?

- **Learning Outcome:** Students will be able to design a website using HTML and CSS.
- **Teaching Activity:** Students attend lectures and ppts for HTML and CSS, and fixed set of lab exercises.
- **Assessment Method:** Students create a functional website that meets specific design requirements, demonstrating their understanding of HTML and CSS.

Answer:

1. **Yes**
2. **No** **WHY?**



Is there a constructive alignment?

- **Learning Outcome:** Students will be able to design a website using HTML and CSS.
- **Teaching Activity:** Students attend lectures and hands-on workshops on HTML and CSS, and practice coding exercises and have completed course projects.
- **Assessment Method:** Students are asked to write how specific elements of UI are designed, demonstrating their understanding of HTML and CSS.

Answer:

1. **Yes**
2. **No**

Is there a constructive alignment?

- **Learning Outcome:** Students will be able to design a website using HTML and CSS.
- **Teaching Activity:** Students attend lectures and hands-on workshops on HTML and CSS, and practice coding exercises and have completed course projects.
- **Assessment Method:** Students are asked to write how specific elements of UI are designed, demonstrating their understanding of HTML and CSS.

Answer:

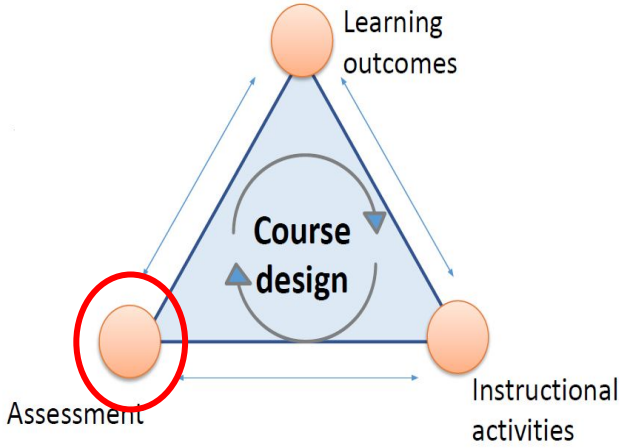
1. **Yes**
2. **No** **WHY?**

Is there a constructive alignment?

- **Learning Outcome:** Students will be able to design a website using HTML and CSS.
- **Teaching Activity:** Students attend lectures and hands-on workshops on HTML and CSS, and practice coding exercises and have completed course projects.
- **Assessment Method:** Students are asked to write how specific elements of UI are designed, demonstrating their understanding of HTML and CSS.

Answer:

1. **Yes**
2. **No** **WHY?**



Bloom's Taxonomy - for writing specific learning outcomes

Level	Description	Action verbs
Create	Combine parts to make (new) whole, creative behaviours, propose plans	design, combine, devise, modify
Evaluate	Judge value based on criteria, decision making	assess, conclude, contrast, evaluate
Analyze	Separate whole into parts until structure of whole and relationship between parts is clear	analyze, infer examine, dissect
Apply	Use knowledge in a new situation. Involves rules, methods, laws, principles	Apply, calculate, solve, predict
Understand	Grasp meaning, explain, interpret, translate, paraphrase	describe, explain, give example
Recall	Recognize, recall facts	define, identify

For More details on learning outcomes

See the PPCCLT channel on YouTube;

Learning objectives - https://www.youtube.com/watch?v=Os5rY2faig8

Bloom's Taxonomy: Computer Science Example question

Level	Cognitive Process	Computer Science Example
Remember	Recall facts, terms	*List the basic data types in Python.*
Understand	Explain concepts	*Describe how a for loop works in a programming language of your choice.*
Apply	Use knowledge	*Write a Python program to implement linear search.*
Analyze	Break down, categorize	*An avid reader wants to buy books, he chooses two books of different price from the store. He decides to buy the book which is less expensive. Write a C program to read and print two book details (i.e. Book Title, Author, Price, number of pages, year of publication) chosen by the reader. Also print the Book title which the reader bought and Book details which is recently published.*
Evaluate	Judge, compare	*"A city is planning to optimize its real-time emergency response system. The city map is modeled as a directed weighted graph, where intersections are nodes and roads are edges with weights representing travel times in minutes." The control center needs to perform an operation: Quickly determine the shortest route from the emergency center (at a fixed node) to any destination node when an incident is reported. For the above operation, you can solve it using Floyd's or Dijkstra's algorithms. Which one would you choose to perform the operation? Justify your answer and write the algorithm.*
Create	Design, produce	*Design and implement a simple stack-based calculator using object-oriented programming.*

Quality of questions in an assessment

Poor

1. Analyze the benefits and drawbacks of using recursion compared to iteration in terms of efficiency, readability, and memory usage.

Good

1. Which method is better suited for solving a given problem—recursion or iteration? Justify your answer by discussing efficiency, readability, and memory usage.

Samples of good questions - Analogy based questions

Analogy: Analogy-based questions are **inquiries** that require students to **draw parallels between different concepts, ideas, or situations.**

These questions often involve **comparing the characteristics, relationships, or functions of one thing to another** through the use of analogies.

Analogies serve as a **tool to help convey a deeper understanding** of a topic by relating it to something more familiar.

In educational contexts, analogy-based questions can be used to **assess critical thinking skills**, **comprehension**, and the ability to **make connections between abstract concepts** and **real-world scenarios**.

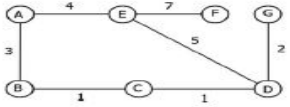
Samples of good questions - Analogy based questions

- *What are the two Graph-traversals ? Provide an* ***analogy*** *that helps elucidate the differences in their exploration strategies and applications.*
- *Explore the characteristics of a hash table and its collision resolution methods. Devise an* ***analogy*** *that captures the essence of hashing and collision handling, drawing parallels to a real-world situation or system.*

Samples of good questions - Scenario-based

Given an undirected weighted connected graph having n nodes and m edges. At time $t = 0$, all nodes are having a **orange** diamond. The diamond in the nodes change their appearance by switching from **green** to **orange** and vice-versa after every '**b**' seconds for all the nodes simultaneously. Payal is initially at node 1. She wants to reach at node **n** in the minimum time possible. The weight of an edge denotes the time taken to travel through the edge. But Payal has to follow the constraint: At any node, if the diamond in that node is **orange** when Payal reaches there, she can pass through that node immediately, else she has to wait at that node till it becomes **orange**. Note: If Payal reaches a node at an instance when the node is switching its appearance, she follows the above-mentioned rules according to the new appearance of the node.

Consider the below graph for an example with b = 4.

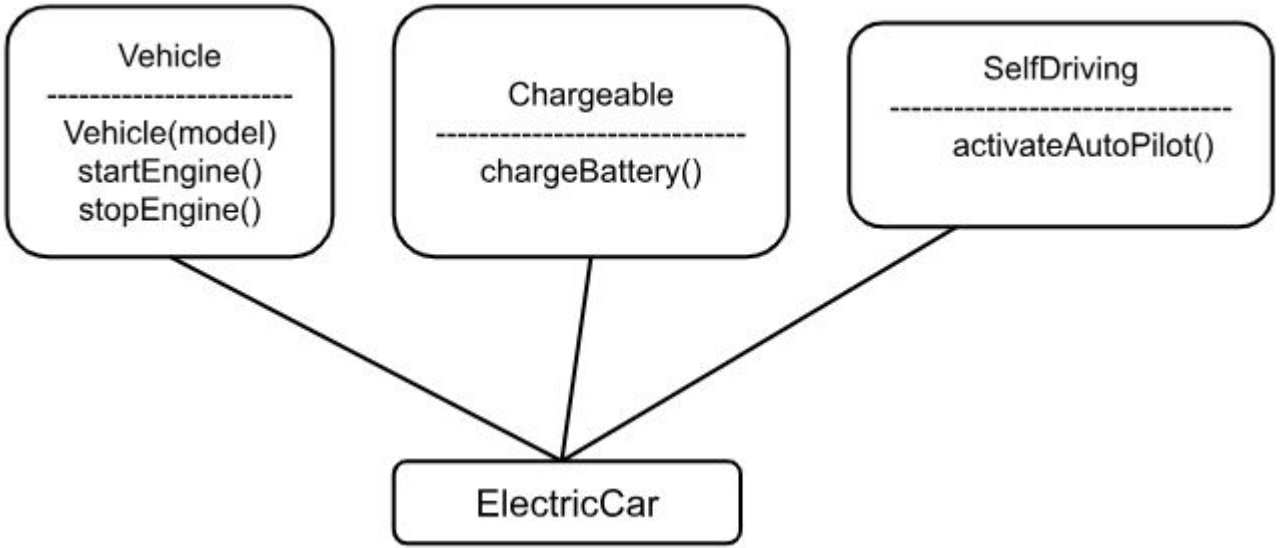


- Node number A: Payal can visit any of its adjacent nodes. She chooses to visit node B. Since the diamond at this node is orange at the 0^{th} second, she can visit node B which takes 3 seconds.
- Node number B: The diamond in this node is still orange since they change color every 4 seconds. Payal now chooses to visit node C which takes 1 second, and we are now at the 4th second.
- Node number C: 4 seconds have passed, and the diamond in the nodes have already become green. Payal has to wait for 4 more seconds until the diamond in the node becomes orange.
- Node number D: Payal takes the route to node D which takes 1 second. So far, 9 seconds have passed.
- Node number G: The diamond in the node is still orange and Payal can go to node G. It takes 2 seconds to reach node G.
- In total it takes 11 seconds to reach node G (the destination). This is the minimum time possible.

(A) Write a pseudocode to find the *minimum time* needed for Payal to travel from Node 1 to Node **n**.

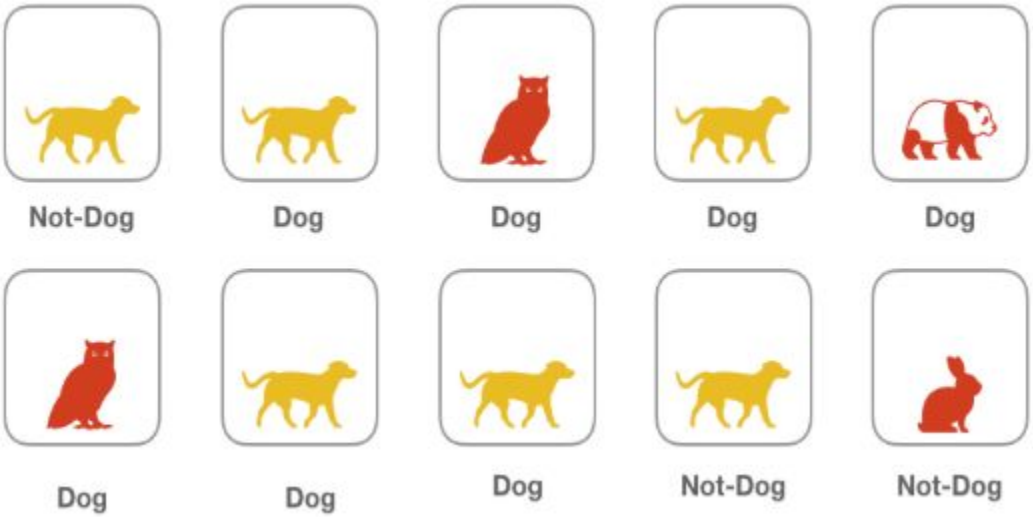
Samples of good questions - Scenario-based

Identify the OOP concept for the given structure below and implement the same in a Java program.



Samples of good questions

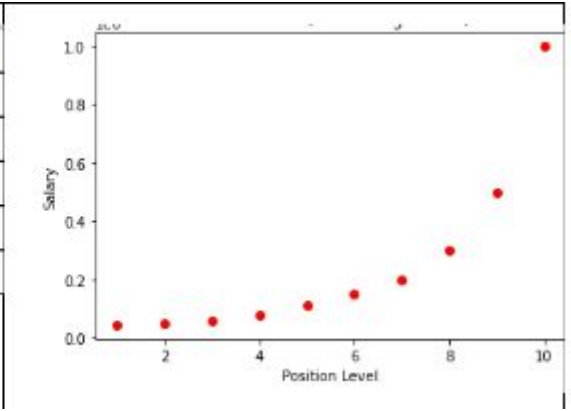
Generate a confusion matrix for the Dog Detector classifier for the below predictions. Find the precision, recall and F1 score.



Samples of good questions

Consider the following sample dataset and its visualization

Position	Level	Salary
Business Analyst	1	45000
Junior Consultant	2	50000
Senior Consultant	3	60000
Manager	4	80000
Country Manager	5	110000



Justify which regression model is suitable for this dataset and write the complete python code for the same. Predicting result for a new instance (Manager,4) .

Samples of good questions - Predict output

```c
#include <stdio.h>
int main( )
{
    char word[100];
    int l=0,i;
    printf("Input the string : ");
    gets(word); // Assume the string entered is "DESSERTS"
    while(word[l]!='\0')
       l++;
    printf("Printing the characters \n");
    for(i=l;i>=0;i--)
        {
          printf("%c  ", word[i]);
        }
    return 0;
}
```

Samples of good questions - Refute

Refute: Refute question prompts students to provide a **counter-example** illustrating that a specified **concept/code fragment is an erroneous solution** for a given task. This type of question serves as an assessment item, aiming to foster the ongoing development of a student's understanding or, alternatively, to showcase their abilities by **revealing their comprehension of both the task and the concept/code.** The significance of these abilities is amplified in light of the rise of generative AI technologies, which have the capacity to generate answers or code that may appear plausible to novice students but is not necessarily accurate in all instances.

Samples of good questions - Refute

The below code snippet is used to delete an element from a doubly linked list (L) from a specific position (k). Refute by providing an input where it does not do its job properly and provide the modified version.

```
function DList-Delete-Position(L, k)
    *node p = head
    while p ≠ k do
        q ← p
        p ← next[p]
    end while
    prev[next[k]] ← prev[k]
    next[k] ← next[q]
    next[q] ← NIL
end function
```

Samples of good questions - Refute

Explain in Plain English question (adapted from [2]): Write a short, English language description of what this function does.

```
1  // Assume array != null
2  // and array.length > 0
3  int f(int [] array) {
4    int x = array[0];
5    int y = 0;
6    for (int i = 1; i < array.length; i++) {
7      if (array[i] < x) {
8        x = array[i];
9        y = i;
10     }
11   }
12   return y;
13 }
```

Sterile vs. meaningful variable and function names

Acceptable answer (*EiPE*)	Acceptable answer (*Refute*)		
	Input `array` / Correct	Option answer	Function's return value
"It returns the index of the smallest value in the array" [2]	(a) {2}	2	0
	(b) {0, 0}	1	0
	(c) Correct	--	--
	For "pure" *Refute*, discard choice (c)		

Refute variant: Choose at most one option that *precisely* explains what this function does. For every other option, provide an input array for which the function's return value is *inconsistent* with this option.

(a) It returns the minimum integer in the given array.
(b) It returns the *largest* index at which the minimum value appears in the given array.
(c) It returns the *smallest* index at which the minimum value appears in the given array.

```
1  // Assume array != null
2  // and array.length > 0
3  int getIndex(int [] array) {
4    int min = array[0];  // minimum so far
5    int index = 0;
6    for (int i = 1; i < array.length; i++) {
7      if (array[i] < min) {
8        min = array[i];  // new minimum
9        index = i;
10     }
11   }
12   return index;
13 }
```

Comments to clarify intent

Course Assessment: Syllabus Coverage and Component-wise weightages

Assessment Component	**Weightage**	**Syllabus Coverage**
One Mid-Term Examination	30%	Initial 60% of the syllabus with equal distribution
Assignments/Case Studies/Mini-Project	**20%**	**Remaining 40% of the syllabus with equal distribution**
Quizzes (Maximum 5)	20%	Equal distribution across all modules
Semester End Examination	30%	Equal distribution across all modules

Which of these strategies can be used for assessment and which are not?

- TPS
- Peer Instruction
- Concept mapping
- Game based Learning
- Cooperative Learning - Jigsaw
- Project - based learning
- Case Studies
- Collaborative Learning - Group Discussion
- Experiential Learning - Hackathon
- Flipped classroom

Write questions for the course you handle this semester

Analogy based -

Scenario based -

Refute -

Predict output -

Analyze level -

Evaluate level -

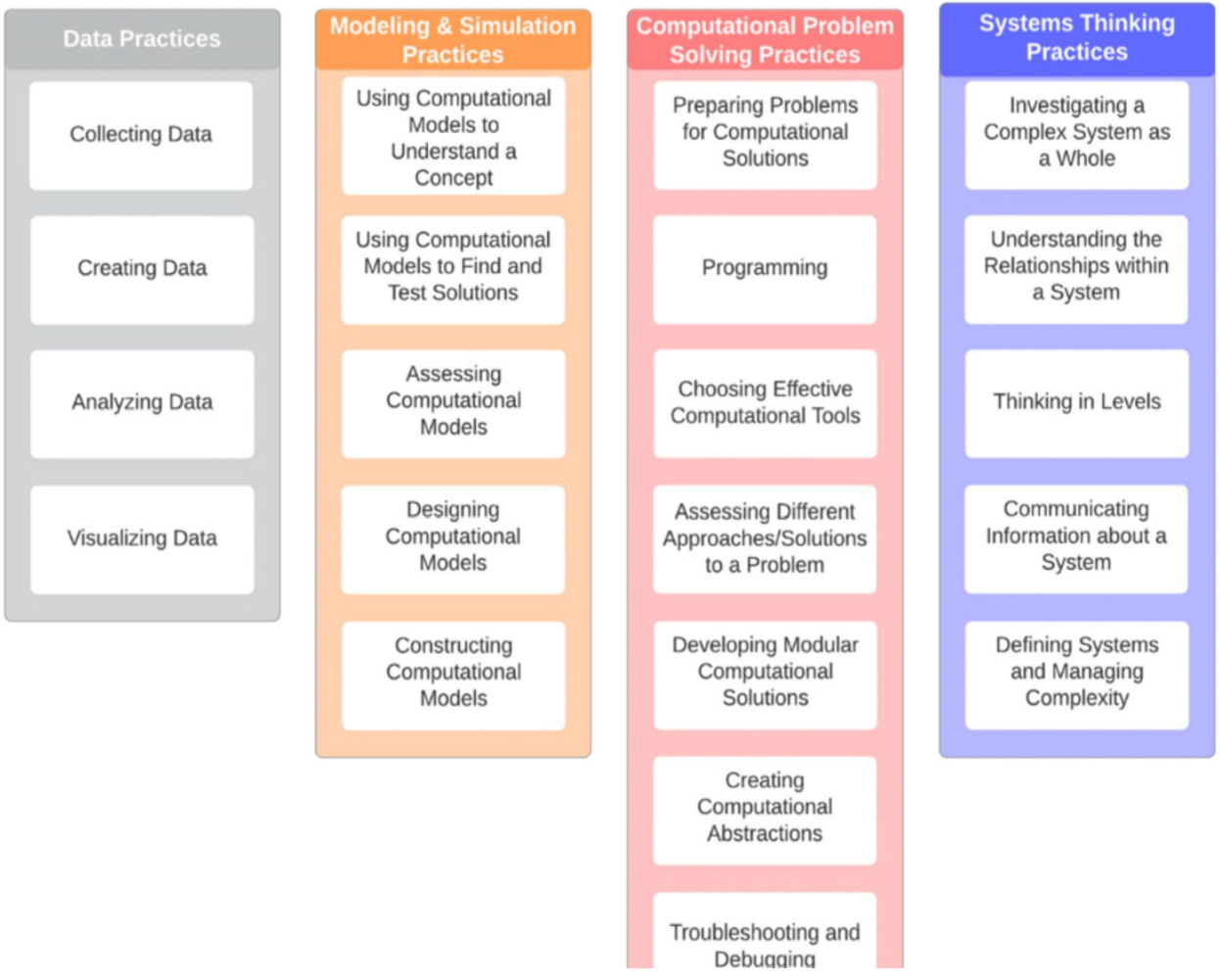
Using AI for generating questions - Discuss the Pros and cons

Rubrics Generator

Magicschool: https://www.magicschool.ai/tools/rubric-generator

Computational Thinking

Taxonomy of Computational Thinking in mathematics and science



Select any one taxonomy and label your paper based on it. Makes notes on gaps and how the curriculum plan can be improved.
You may combine elements from other taxonomies as well

- Criticism that Bloom’s taxonomy is not mapped to CS skills
- Focused on skills to be developed by learners of computer science
- **Resource**: https://tinyurl.com/yxwssac7
- https://www.mdpi.com/2073-431X/14/5/192

Autograding

Empowering CS Teachers for Scalable Assessment

Why Autograding?

Scalability: Grade hundreds of submissions automatically

Consistency: Remove human grading variance

Immediate feedback: Help students learn faster

Ideal for coding, data structures, ML, etc.

For Github classroom: You need to enrol yourself as a teacher in the Github classroom before getting autograding capabilities

https://education.github.com/globalcampus/teacher

Using one or more of these tools, evaluate your students' submissions

- https://www.gradescope.com/ (Python, Java, C++ etc)
- https://tinyurl.com/muj7nvvj Autograder.io
- replit.com (Replit for teams for education)
- nbgrader(for Jupyter notebooks)

  https://github.com/jupyter/nbgrader
- Github Classroom + Github Actions https://classroom.github.com

Deliverables

1. Installation - 15mins
2. Grading scheme - 15 mins
3. Create five complex coding problems - 15 mins
4. Create unit test - pytest/junit/unittest - 15 mins
5. Automate the process - 10mins