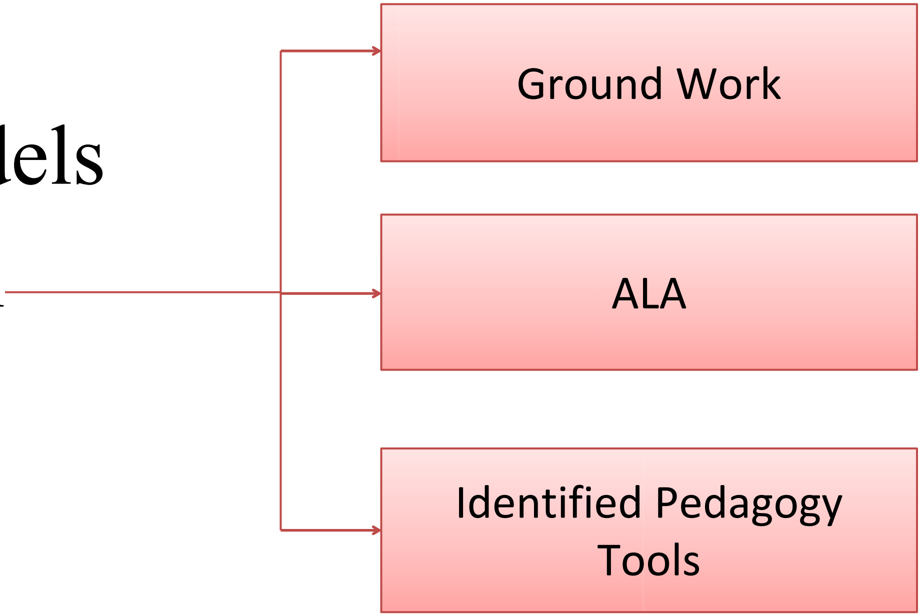
Active Learner Approach (ALA) for Teaching Programming related Subjects

Authors
Ms.N. Ravitha Rajalakshmi,
Dr. B. Sangeetha,
Assistant Professor (Sr. Gr.),
Department of Information Technology,
PSG College of Technology

Agenda

- Introduction
- Objective
- Related Models
- Contribution
- Conclusion



Ground Work

ALA

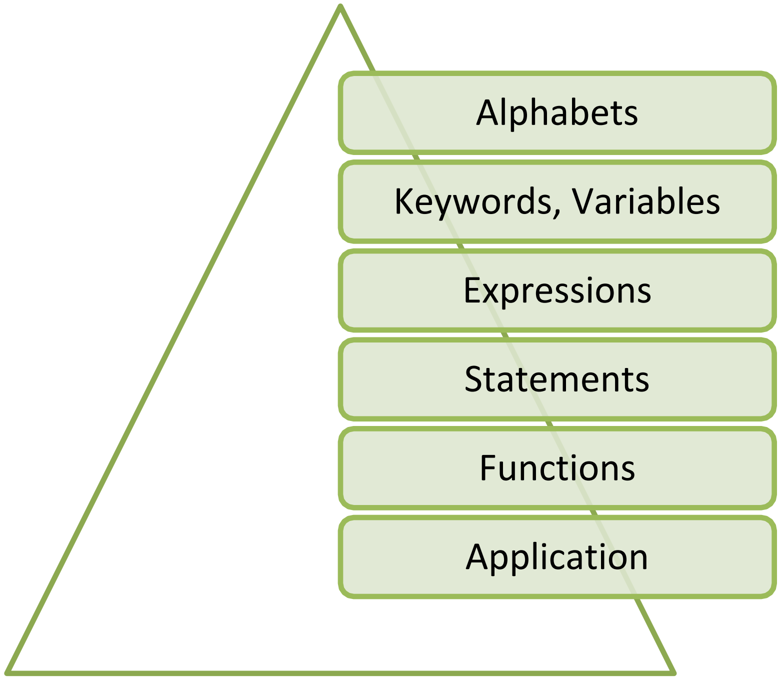
Identified Pedagogy Tools

Introduction

- Is the Journey of Learning in the current scenario a wholesome experience?
- Statistics Reported by AICTE – 60%
- Observed Reasons
    - Skill Gap
    - Lack of Motivation
    - Fear for Learning
- Measures to alleviate the problem?

Strategies

- Strategies
  - Top-down approach
  - Bottom-up approach



Alphabets

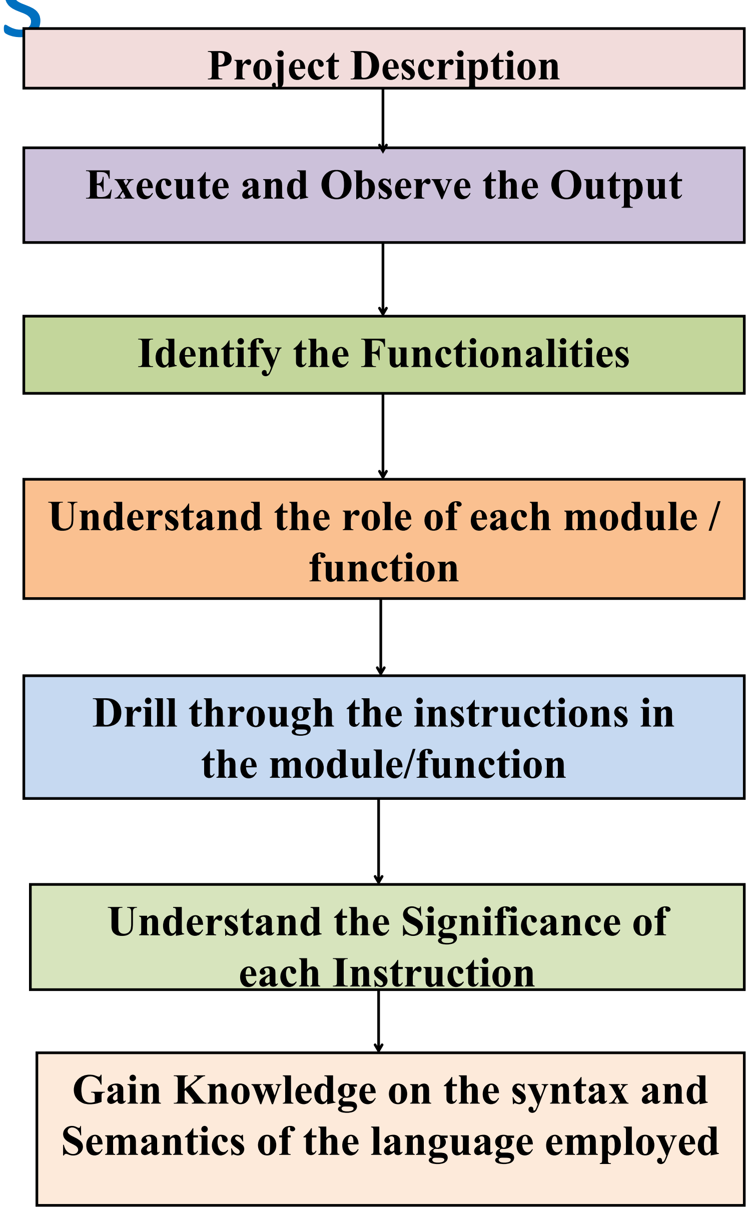
Keywords, Variables

Expressions

Statements

Functions

Application



Project Description

Execute and Observe the Output

Identify the Functionalities

Understand the role of each module / function

Drill through the instructions in the module/function

Understand the Significance of each Instruction

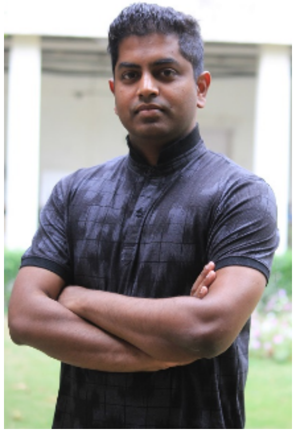
Gain Knowledge on the syntax and Semantics of the language employed

Related Models

- Existing works are primarily grouped under four categories:
  - Learning model
  - Instructional Techniques
  - Learner Engagement
  - Pedagogy practices

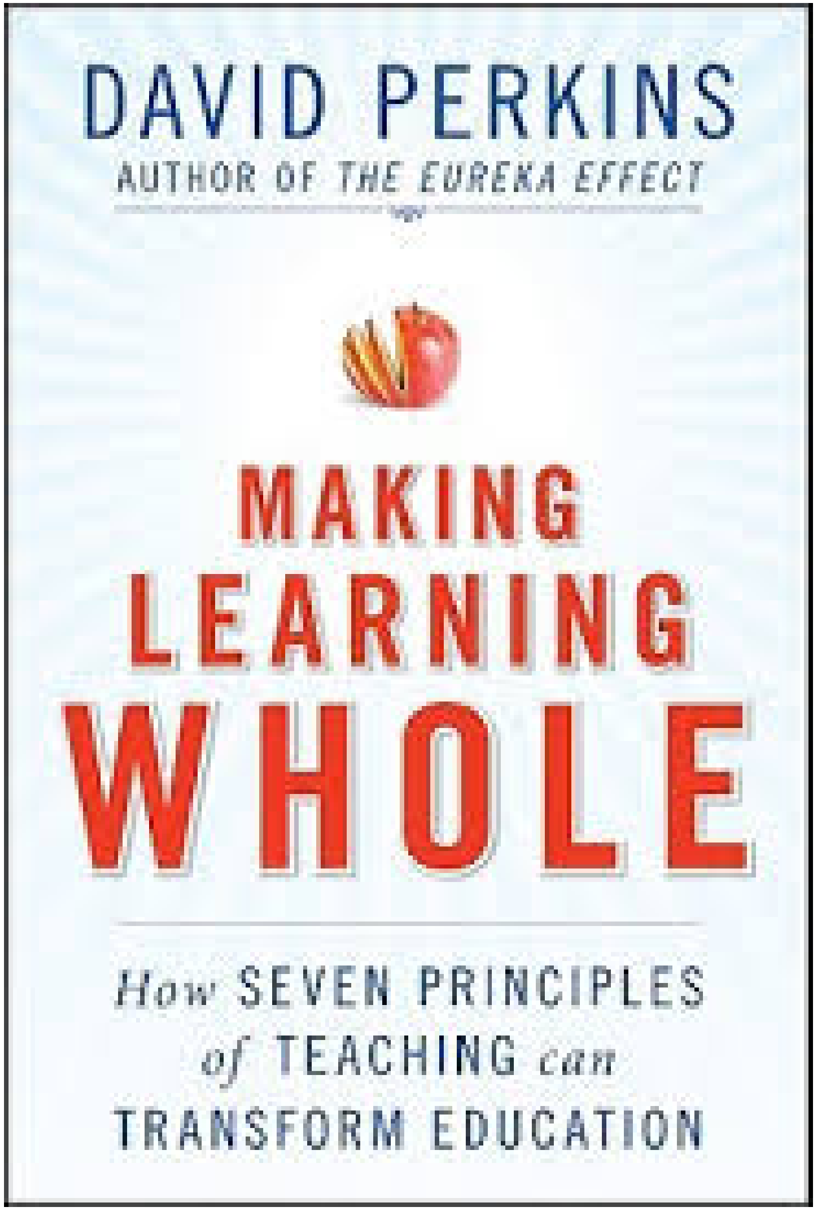
NPTEL

Joy of computing using python



Deep Learning with Fast.Ai

fast.ai



DAVID PERKINS

AUTHOR OF *THE EUREKA EFFECT*

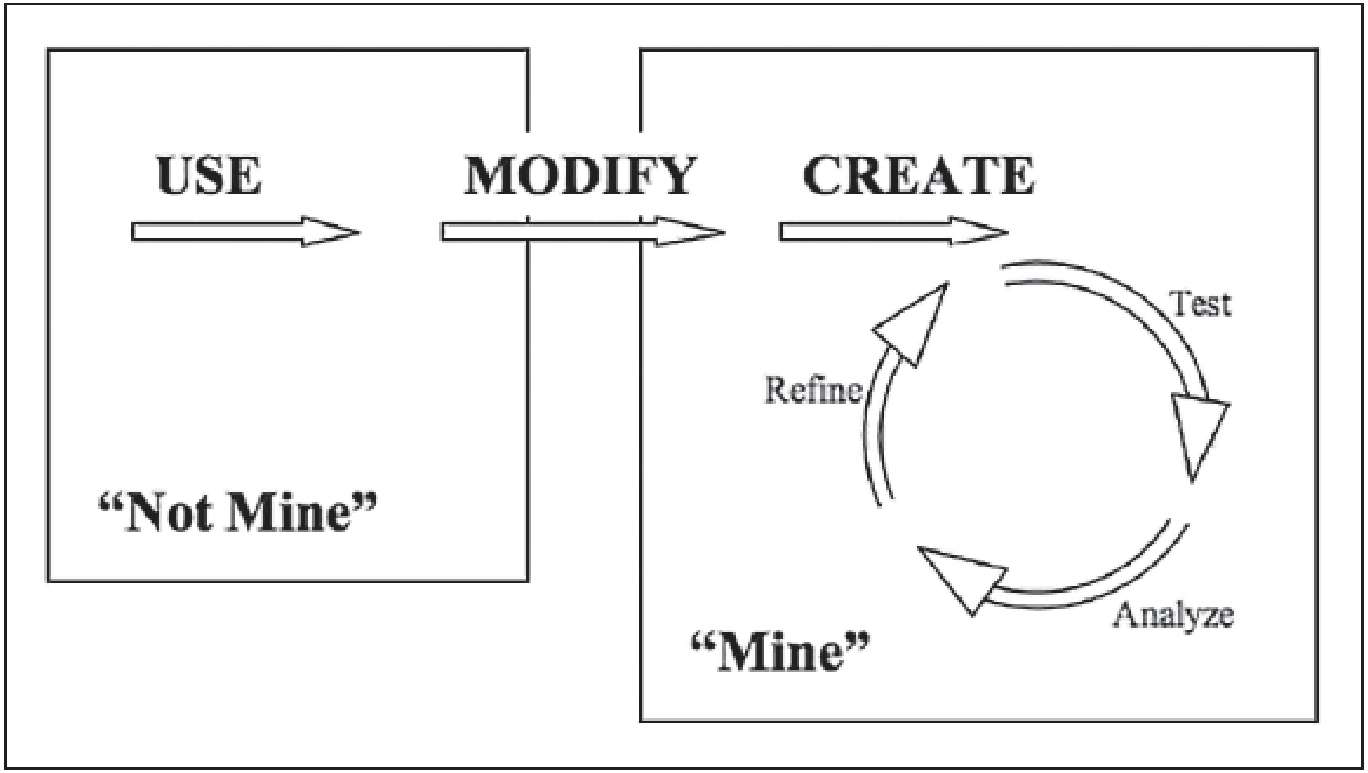
MAKING LEARNING WHOLE

How SEVEN PRINCIPLES *of* TEACHING *can* TRANSFORM EDUCATION

The whole journey of learning can be effective only if the player/learner get his hands dirty with the game. He adds that it doesn’t require the player to be aware of the rules of the game before playing.

1. Play whole game
2. Make the game worth playing
3. Work on hard parts
4. Play out of town
5. Play hidden game
6. Learn from team
7. Learn the game of learning

ALA is inspired from the use-modify-create model and it enhances it by adding learner engagement.



Use-Modify-Create Model (Lee et al, 2011)

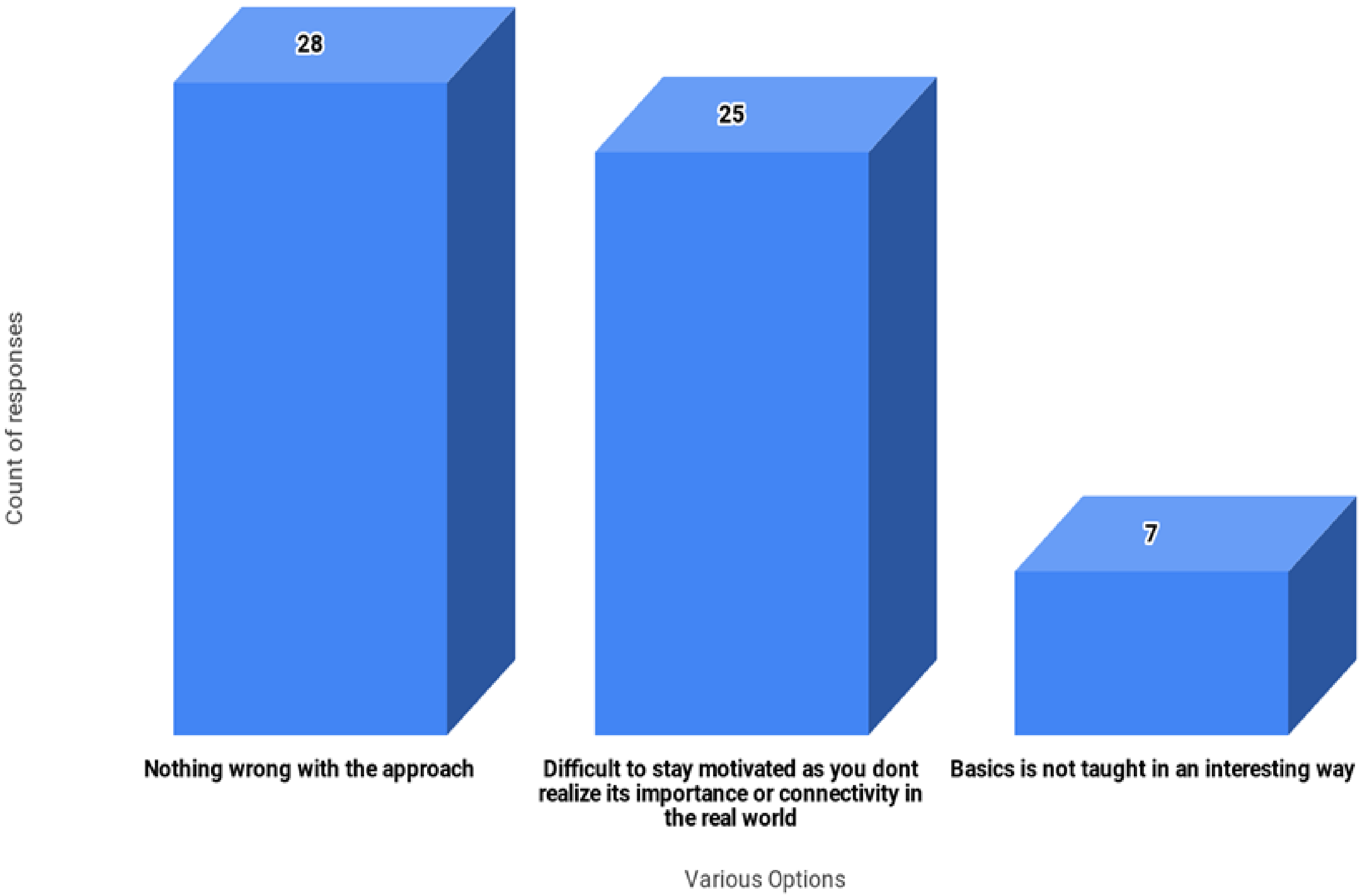
Contributions

- **Extensive survey** with questionnaire to obtain suggestions from the learners
- Proposed **ALA (Active Learner Approach)**
- Identified **pedagogy tools** to support the approach

Survey Questionnaire

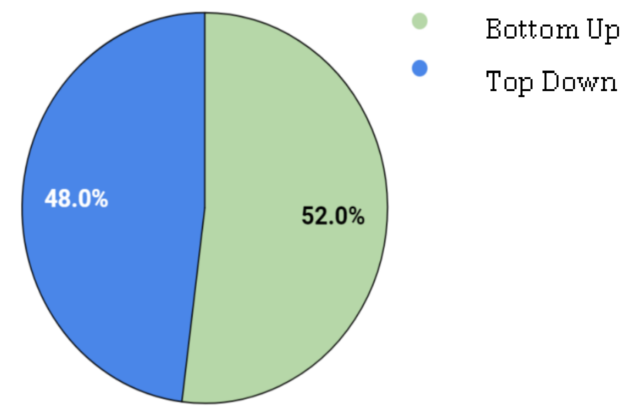
Fixed Response Questions	
1.	From your perspective, how can programming be taught? So that its effective
2.	What is lacking in the bottom up teaching strategy?
3.	Were you been able to transform or contribute to projects after taking the programming course?
4.	Rate the modularity of the code written by you.
5.	How do you rate your programming skills in the scale of 1 to 10? 1 indicates low and 10 indicates high
6.	What was your stream in the 12th standard?
Open Ended Questions	
7.	What teaching software's (or) tools can be used to improve the teaching process (Scratch, Alice)?
8.	Have you referred to any other websites (or) learning resources to improve your programming skills? If so, Mention them
9.	What difficulties did you face during the programming course?
10.	What style or coding standard do you employ?

Survey Responses for Question 2 (What is lacking in the bottom up teaching strategy)

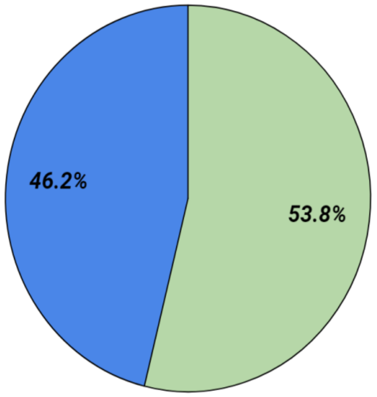


Responses for Effective Strategy

CS students

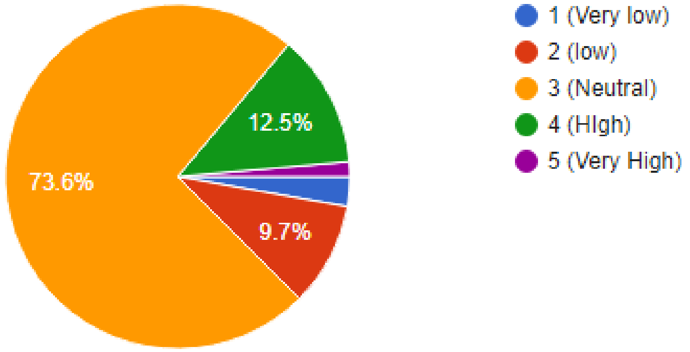


Biology Students



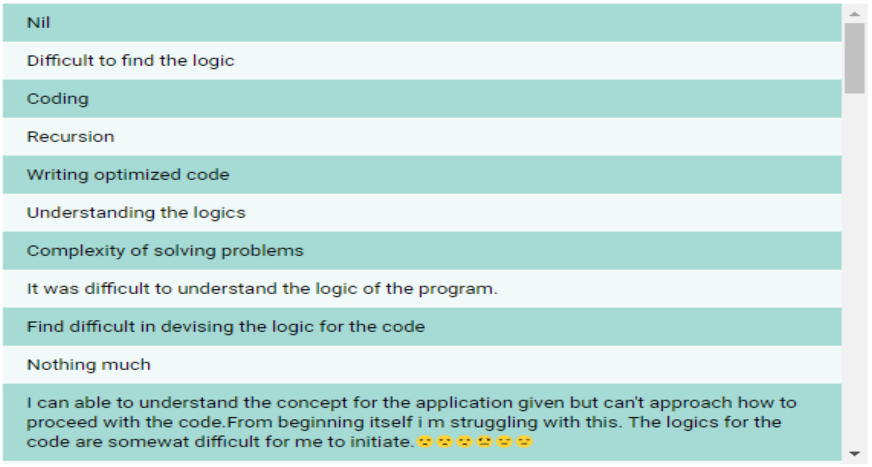
Rate the modularity of the code written by you ?

72 responses



What difficulties did you face during the programming course?

72 responses



Nil

Difficult to find the logic

Coding

Recursion

Writing optimized code

Understanding the logics

Complexity of solving problems

It was difficult to understand the logic of the program.

Find difficult in devising the logic for the code

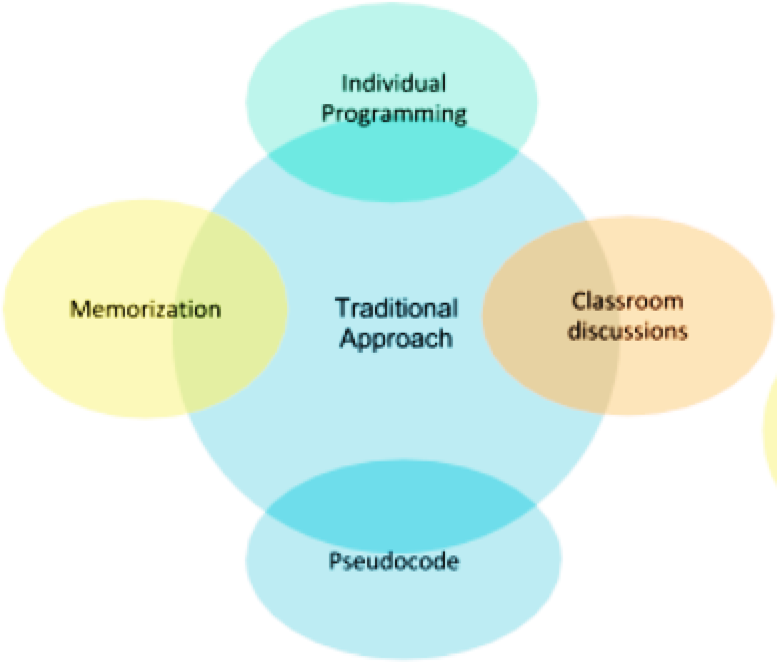
Nothing much

I can able to understand the concept for the application given but can't approach how to proceed with the code.From beginning itself i m struggling with this. The logics for the code are somewat difficult for me to initiate.😔😔😔😔😔😔

Model Aspects

- Demonstration
  - Learn from Experience
- Evaluation
  - Learn from activity
- Learner Engagement
  - Learn from peers

Pedagogy Tools



Individual Programming

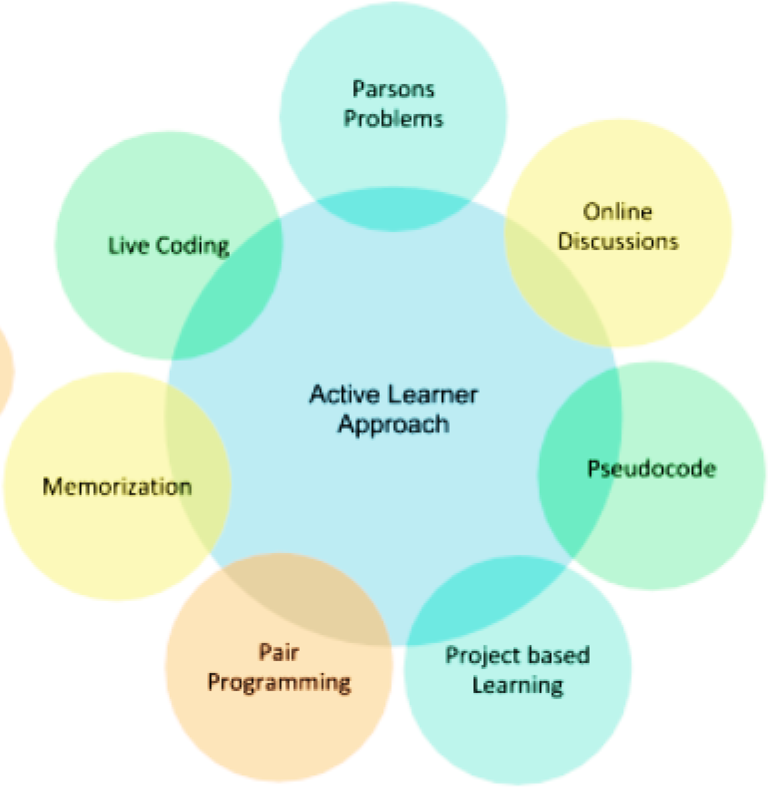
Memorization

Traditional Approach

Classroom discussions

Pseudocode

a) Traditional Approach



Parsons Problems

Live Coding

Online Discussions

Active Learner Approach

Memorization

Pseudocode

Pair Programming

Project based Learning

b) Active Learning Approach

Conclusion

- Proposed ALA model will improve the TLP
- Motivate the students and make themselves engage in life long learning
- Equips them Industry-Ready

References

- Perkins, David.N. (2009) *Making Learning Whole.* San Francisco, CA: Jossey-Boss.
- Linderholm, Carl.E (1971) *Mathematics Made Difficult*. London: Butler & Tanner Ltd.
- Parasuraman, A.(1991) *Marketing Research - 2nd Edition.* New Delhi:Addison-Wesley Publishing Company, Inc.
- Waite, J.(2017) *Pedagogy in Teaching Computer Science in Schools: A Literature Review. London.* Supplement to Royal Society Computing Education Project Report.
- Garneli, V., Giannakos, M. N., & Chorianopoulos, K. (2015) Computing education in K-12 schools: A review. Global Engineering Education Conference (EDUCON), 2015 IEEE (pp. 543 - 551). IEEE.
- Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., et al. (2011) Computational thinking for youth in practice. *ACM Inroads*, 2(1),32–37.
- Corney, M., Teague, D., Ahadi, A., & Lister, R. (2012). Some empirical results for neo-Piagetian reasoning in novice programmers and the relationship to code explanation questions. *Proceedings of the Fourteenth Australasian Computing Education Conference*-Volume 123(pp. 77–86). Australian Computer Society, Inc

References

- Teague, D., & Lister, R. (2014c). Programming: reading, writing and reversing. *Proceedings of the 2014 conference on Innovation & technology in computer science education(*pp. 285–290). ACM.
- Grover, Pea, & Cooper. (2015). Designing for deeper learning in a blended computer science course for middle school students. *Computer Science Education*, 25(2), 199–237.
- Williams, L. A., & Kessler, R. R. (2000). All I really need to know about pair programming I learned in kindergarten. Communications of the ACM, 43(5), 108–114.
- Rubin, M. J. (2013). The effectiveness of live-coding to teach introductory programming. *Proceeding of the 44th ACM technical symposium on Computer science education* (pp. 651–656). ACM.

Thank you