Self-Teaching Evaluation – COP 3530 Data Structure (Effort Intensive Course)

-Janki Bhimani

Objective –

The study aims to comprehensively assess the learning experience and progress of participants in a C++ programming course across multiple modules. Through a series of inquiries, the survey covers key aspects such as identifying the pillars essential for effective learning, evaluating time management in completing module tasks, gauging confidence in C++ programming skills, and understanding participants' preferences for learning content. Additionally, participants are asked about the challenges they faced in understanding specific topics, their ability to design algorithms, and the time spent on different modules. The survey concludes by exploring the overall impact of the course on participants' learning, the inclusivity of teaching video lectures, and their satisfaction, along with reflections on the most challenging module and the potential benefits for future courses. The objective is to obtain a holistic understanding of the participants' experiences, challenges, and perceptions throughout the course in each module and keep improving continually.

Sample Questions –

1. **Key Pillars of Effective Learning:**
    - "What are the key pillars which will help you get the most out of this class? (may pick more than one)"
2. **Time Management and Module Understanding:**
    - "Did you have enough time for all the items in the to-do list of this Module?"
    - "How much time did you spend to finish the ToDo items of this Module?"
3. **Challenges and Confidence in Learning:**
    - "Which topic did you find the toughest?"
    - "Are you confident that you will be able to program well in C++ for your coding assignments and projects of this course?"
4. **Learning Preferences:**
    - "What form of content did you find most useful to learn?"
5. **Algorithmic Understanding and Application:**
    - "Can you design your own algorithms from your own logic to solve problems using the concepts of this Module?"
    - "Can you write a pseudo-code on your own from an algorithm?"
    - "Can you write a C++ code on your own from the pseudo-code?"
6. **Overall Reflection on Learning:**
    - "Did you make progress towards the teaching pillars of this course?"

- "Do you think the scores you got on your assignments, quizzes, and projects reflect evidence of your learning?"

7. **Inclusivity of Teaching:**
    - "Did the course have inclusive teaching video lectures that helped you to learn better?"
8. **Course Satisfaction and Future Outlook:**
    - "Overall, are you happy about taking this course in C++ and do you think this will help you to excel in upcoming courses such as Operating System?"

Summary of Responses –

The survey results illuminate a diverse array of strategies employed by instructor to enhance their learning experience in the class. Key success pillars, including a preference for hands-on experience, teaching to learn, and prioritizing understanding over memorization, emerge as crucial factors. The time allocated to tasks in each module varied, with most students investing approximately 6 hours. Effective time management is evident, as a majority reported having ample time for the to-do list, displaying a range of time investment. Notably, the practice of watching videos more than once for better comprehension and quiz results was not widespread. Participants exhibited confidence in writing pseudo-code for algorithms and translating it into C++ code, maintaining high levels of programming proficiency throughout the modules.

Delving into the Data Structures course, certain topics posed consistent challenges for students. The Vector Container and Pointers were notably identified as difficult, indicating a shared struggle. Exploring searching and sorting algorithms revealed Heap Sort, Binary Search, and Bubble Sort as particularly challenging. While participants demonstrated conceptual understanding, mastering specific sorting methodologies presented persistent challenges. In the domain of algorithms and data structures, some students found topics like queues, priority queues, and linked lists more demanding. However, confidence in programming skills remained robust for the majority. Instances were noted in backtracking, recursion, and problem-solving where specific tasks, such as those involving Sudoku or N Queens, were perceived as more demanding. The survey underscores the necessity for tailored teaching methods to address these challenges and emphasizes the ongoing support required to bolster confidence in C++ proficiency.

The responses to the survey questions provide valuable insights into participants' experiences in the C++ programming course. The majority acknowledge progress toward teaching pillars, appreciating the professor's clear teaching style and opportunities for clarification. While concerns about project toughness exist, the value of peer interactions is evident. The impact of the COVID-19 pandemic on participation and collaboration is recognized during certain semesters. The implemented feedback system receives positive acknowledgment, fostering a healthy learning environment. Some students actively engage in self-directed learning, using textbooks and online resources to enhance understanding. Diverse approaches to learning, including hands-on coding and varied preferences, are highlighted. Overall, participants express satisfaction, finding joy in coding assignments and appreciating the instructor's responsiveness, emphasizing the importance of clear communication and interactive learning methods for a positive and effective learning experience in the course.