

Course & Section: CMN 001 A18

Assignment: Persuasive Speech Outline

Name: Sriram Suresh

General Purpose: To persuade

Specific Purpose: To persuade audience to learn to code and think computationally

Target Audience: Mostly Neutral and Hostile

Attention:

You may be surprised to know that you already think computationally, i.e., like a computer, from time to time. However, you can maximize the benefits of computational thinking in your life by recognizing the instances in which you use it and gaining a deeper understanding into how to use it more often. Whenever I try to convince who doesn't code to start coding, I mostly get hit with the "But coding is boring, and I don't think I can work a 9-5 staring at and writing obscure looking code". However, these machines can benefit you much more than just allowing you to play video games and write essays for your classes. Taking my own example, I have come to realize that the way I think about real-life situations has changed drastically ever since I started to learn to code and think computationally.

Need:

1. Often in one's day, one encounters complex situations which might require extensive and complicated thinking to get through.

- a. For example, if you are in a soccer team and you want to gain the upper hand on your opponents for a crucial match by analyzing their gameplay.

Transition: Now that we have seen how some daily situations could be troublesome and complicated, let us look at how computational thinking might help overcoming these situations.

Satisfaction:

1. Computational thinking teaches you to recognize patterns in real-life situations and use the “algorithms” you know to help you make the best out of a situation
 - a. Looking at the soccer example from before, now you could follow an “algorithm” to understand and decode their strategies and use this information to your advantage.
 2. Computational thinking helps you break down real-life problems into sub-problems and then tackle each of these sub-problems with an effective method to get optimal results.
 3. Computational thinking allows you to create a list of instructions for you to follow and with this, you have a set of steps to work through a problem, rather than work aimlessly.
 4. Coding has helped me communicate more effectively. Since you learn to speak to the computer using a programming language, learning to code teaches you to communicate using simple but effective terms.
- Using the skills computational thinking provides you with mentioned above, you can come up with solutions to a variety of real-life situations.

Transition: Now that we have seen how computational thinking can help tackle complicated everyday situations, let us look at some concerns you may have about starting to code and think computationally.

Visualization:

1. Audience objections:
 - a. The audience may feel that it is inconvenient to pick up such a skill when they are so busy with their other schoolwork especially when there is no obvious relevance to their field of study.
 - b. If persuaded to start coding and thinking computationally, the audience may be confused and might feel a little lost on where to begin with teaching themselves to think computationally.
2. Starting to reason about your daily life situations with a computational thinking mindset, i.e., by recognizing patterns, breaking down problems into subproblems, etc., will help the audience come up with solutions to their problems easily. With more practice on using these principles, they will start solving real life problems much faster.

Action:

To start learning to code, you can start even by watching YouTube videos on the basics of coding. You should remember that here, you are not learning to code to work a mundane 9-5 job,

but instead, to start incorporating computational thinking into the way you process the decisions you make every day. In conclusion, I hope that after listening to the points I made on why you should learn to code and think computationally, you are now aware of how these principles could help you tackle real-life situations with ease, much like a computer does.

Works Cited:

- Victoria, Katie. “What is Computational Thinking? Why thinking like a computer builds skills for success”, *teachyourkidscode.com*, Dec 29th, 2018, <https://teachyourkidscode.com/what-is-computational-thinking/>
- Cummins, Kevin. “Five reasons why computational thinking is an essential tool for teachers and students.”, *innovativeteachingideas.com*, December 6th, 2020, <https://www.innovativeteachingideas.com/blog/five-reasons-why-computational-thinking-is-an-essential-tool-for-teachers-and-students>
- McVeigh-Murphy, Anna. “What Is Computational Thinking? And Why Is It Important for Students?”, *equip.learning.com*, May 22nd, 2019, <https://equip.learning.com/computational-thinking>
- Pinder, Nick. “Why You Should Integrate Computational Thinking Into Your Curriculum”, *iste.org*, Jan 14th, 2022, <https://www.iste.org/explore/computational-thinking/why-you-should-integrate-computational-thinking-your-curriculum>