**Arun Agarwal**

Ben Alfonsi

Final Exam

Introduction to Computer Science

Summer 2018

**Prompt:**

Imagine that you are a college computer science teacher.

You have a diverse set of students; for example, retired folks just looking to a take a course; high school students that want exposure to computer science to prep them for college; professional programmers and software developers; gamers, and “mathletes.”

Your mission is to design an **Introduction to Computer Science** course.

To accomplish your mission, you must identify five (5) main topics that you think should be part of your course.

* Explain why you chose these five (5) topics.
* For each of your (5) topics, make sure you have a clear and brief lesson plan on how you intend to present and explain the topic.
* Additionally, for each of the five (5) topics, explain what homework assignment you would assign to reinforce the topic.

**Your final document should be at least 500 words.**

**ANSWER IS DOWN BELOW!!!!!!!**

**(1,991 WORDS)**

**Answer:**

If I were a college computer science teacher, there would be a lot of things that I would have to consider. Having a diverse set of students, each with varying desires and needs, I need to make a course plan that best accomplishes all of their goals as well as my own goals. Although, since this is an introductory course to Computer Science, I must remember not to stress them out by, for example, starting with advanced program languages. Instead I should start with the basics of Computer Science and push them towards a greater understanding of the course that will help them with future endeavors. Five topics I would then focus on include: what is computer science and what are the basic definitions associated with it; what are the sub-branches available in Computer Science (as well as the details available about them); what is programming and the basics one should understand before starting to program; how to program C and other languages (only the basics); and finally how does computer science play a role in our future. These five topics together will help cover the interests of all my students, whether they are just looking to take a course or learn something for future courses. I hope for my course to be engaging, informative, and most of all fun to guide people to a brighter future.

First, it is important to teach what computer science is and the basic definitions associated with it as a lesson—also as the first lesson—because this information creates the foundation for all other computer science information that will be taught later. One cannot understand how to program without knowing what a computer is first. Even if they already know what a computer is, one would still need to know what programs to use to be able to program and know why they are programming. They cannot learn this information without an introduction of basic information of computer science. This will be the first thing that I teach them—probably on the first class—after introducing myself, stating my background, meeting all the students, and going through the syllabus. I will start by asking the close why are you here. I will listen to everything they have to say, and if they do not say a lot, I will push them to say more with more questions. I want them to express themselves and understand why it is that they are here today. Some of them will need time to think of their answers while others will know right away, and this is something I will keep in mind. After this, I will ask each of them to write down what they think the definition of computer science is, re-read it, and then share it with their neighbor once they think it is as perfect as they can make it. They will keep talking to more and more people until the whole class has agreed upon one definition. I will have them share it, tell them if they were correct, and then share the actual definition. I will then describe what computer scientists do, how you become one, what are the necessary skills, why this job is important, what are the salaries, and how to advance to higher career positions. I will share this information through a video from YouTube, the Bureau of Labor statistics website on computer scientists (they will see the website on the projector and their individual PCs once they pull it up), and by explaining information as it pops into my head. I will then go to Wikipedia to share definitions of important words in Computer Science, such as algorithm, binary system, developer’s environment, and more. I will end with a fun Kahoot that I will prepare beforehand to end on a fun note, see how much they learned, and make everyone more comfortable with one another. Finally, for homework, I would have them make flashcards of all the different words we learned today and study them, so that they will be prepared for a quiz in the beginning of the next class.

For the second class, I would focus on the sub-branches in Computer Science (Web Developer, Software Developer, Systems Analyst, etc.) because this will demonstrate how large of a field Computer Science actually is. By seeing all the careers available in Computer Science, they have a higher chance of finding one they like and possibly majoring in for the future. This would accomplish the main goal I have for this course. It would also show them that computer science is not all about programming and making games like people make it seem. I would start with the quiz I mentioned earlier to test their knowledge (and if applicable, see which students are responsible and which ones are aren’t). I would then turn on the projector and go to computerscience.org. I would tell everyone in the class to as well on their own PCs. I plan to spend the majority of class on this website looking into all the details of each field in computer science (with breaks in between) to appeal to everyone’s likes and desires. I would go into what that field is, how it relates to computer science, how do you major in that field, what are the necessary skills, what are the salaries, and how do you advance in that career. I can find all this information on computerscience.org and use videos to reinforce the information (students pay more attention to videos than a teacher talking for two/three hours). After going through all this, it would be time to go, and I would tell each of them to choose one field they think is really interesting that we went over and to make a Kahoot for that specific field (homework). I will make sure everyone picks a different field (via email), and we will all play the Kahoots next class. This will not only serve as a review, but it will get the class engaged and wanting to go the next class.

Next, for the third class, I would move into the basics of programming (what it is, why it is important, etc.) to give them a pre-class instead of jumping right into the one of the hardest thing they would have to do in computer science. This also serves as semi-break; they have been doing a lot of learning inside and outside of class up till now, and it is good to slow down occasionally. I will still be teaching them important information, and they will not be feeling rushed to learn that information. I will start with the Kahoots that they made for homework. I will put a grade in for effort and completion for the Kahoots (unfortunately I will give zeros to those who did not complete it to show them I am not one to mess around with). After we play everyone’s Kahoots, we would take a short break and move in to programming. I will go onto a trustful website (possibly bls.gov or code academy) and display the websites on the projector. They would, again, open the sites up on their PCs as well to keep them engaged. I did not mention it earlier, but I will constantly monitor their computer screens and their focus by walking around as I am teaching and calling on people for questions. We would go through the website together and learn definitions and basic information on how to program. We will watch a couple of demos to end the class and prepare them for next class. For further preparation, I will have them use resources (such as Scratch) that allow the students to get a taste as to what coding is like (homework). I want them to explore these resources for at least two hours before the next class.

Moving on, for the fourth class, I will be doing some of the most challenging material: programming. Even though this is an introductory course, it is still focused on computer science, and programming plays a major role in computer science. I would be a failure of a teacher if I did not show them anything about how to program. I understand if it seems rushed to get into programming so soon, but at least they will be well prepared for any future plans in computer science. I will start simple on Scratch. Even though they already looked at it for homework, they probably did not realize/play with the higher-level functions available on the program, and I will go through these with them (projector and PCs will be used again). After that, we will go onto a C compiler where we can start making and test code. We will play around with the compiler, I will demonstrate how it works, and then we will write some code. I will go through the “Hello World” demo with them, and I will ask them to write it again without looking at any external resources. Once they successfully complete this, they can plug their earphones in and watch a demo video—one that I already picked out—to learn some other basics programs they can practice. I will answer any questions they have (they will most likely have a lot), and this will take us to the end of the class. I will have them write a program that adds and subtracts two numbers with a C compiler for homework. This will demonstrate their ability to apply what they learn and provide them with a challenge. Those who find this task interesting will make the code over and over until they can write it in the fewest lines of code they find possible.

For the fifth and final class, we will take a broader look at computer science. After learning all the basics of computer science, the students will most likely be asking why. Why do they need this information? What will this career help with in the future? It is important for students to understand this or all interest in the field might dissipate. This lesson would also provide a good conclusion to this course and give them some perspective for the future. It will, hopefully, lead them to a brighter future. Before we talk about this though, I will examine the programs they wrote and put a grade in for each of them based on effort and completion. They will then take their final exams (this is still a college course, so it needs a final exam) to test how much they learned. This exam will focus on all the material we covered from definitions to fields to salaries. It will be mostly multiple choice and end with a free-response, asking them if they are any more aware of what they want for their future after taking this course. After they are done their exams, we will then move onto the lesson I stated earlier. I will explain this based on the information I already know and understand (I should already know this information or else I would not be qualified to teach this course), and by displaying videos describing the future with computers. I will not give them any homework as this is the last class, but I will encourage them to keep practicing programming languages, expand their skills to other coding languages, and continue to consider their interests for the future.

To conclude, as a computer science teacher, the five main topics I would focus on include: what is computer science and what are the basic definitions associated with it; what are the sub-branches available in Computer Science; what is programming and the basics one should understand before starting to program; how to program C and other languages; and how does computer science play a role in our future. I hope my course will help them learn all of this in a fun and engaging way that will guide their choices in the future.