**Week 14**

Literacy refers to the ability to read and write, but it can also refer to a broader set of skills and knowledge related to a particular domain. For example, media literacy refers to the ability to access, analyze, evaluate, and create media content. In this sense, literacy can be seen as a set of competencies and practices that enable individuals to engage with and participate in different aspects of society.

Computer programming can be seen as a literacy because it involves a set of skills and knowledge related to creating, designing, and building digital systems. Programming involves the use of programming languages, which are used to create computer programs, applications, and websites. To be proficient in programming, individuals must not only be able to read and write code, but also understand programming concepts, algorithms, and data structures.

Calling computer programming a literacy highlights its importance in today's society, where digital technologies are increasingly central to many aspects of life. The ability to program enables individuals to create and shape technology, rather than being passive consumers of it. Programming skills are also in high demand in many industries, making them valuable assets in the job market.

However, it is important to recognize that programming literacy is not equally accessible to everyone. Barriers to access can include limited access to technology and resources, inadequate education and training, and lack of diversity in the tech industry. As such, calling programming a literacy should not be seen as a way to exclude those who do not have access to it, but rather as a way to promote greater equity and accessibility in the digital world.

After reading the assigned texts, several new thoughts arise about the notion of coding literacy. One key idea that emerges is the importance of thinking beyond technical skills and viewing programming as a form of communication and cultural production. As Vee (2017) argues, understanding programming from the perspective of literacy "shifts our understandings of both" programming and literacy, making both more capacious. This perspective highlights the importance of understanding the social and cultural dimensions of programming and recognizing that programming is a form of expression and creativity.

Another thought that emerges is the potential for live coding to challenge traditional notions of authorship and ownership in programming. As Blackwell et al. (2022) argue, live coding is a performative and improvisational practice that challenges the idea of programming as a solitary and fixed process. By emphasizing collaboration, interaction, and experimentation, live coding opens up new possibilities for engaging with programming as a cultural practice. This perspective raises important questions about the politics of programming and the ways in which programming can be used to shape and reflect cultural values.

Finally, the idea of programming as performance, as discussed by Aaron (2016) in the video, highlights the importance of creativity and expression in programming. Aaron argues that programming can be seen as a form of musical performance, with code serving as the instrument. This perspective emphasizes the importance of understanding programming as a form of cultural production that can be used to express ideas, emotions, and experiences.

Overall, the readings problematize the notion of coding literacy by highlighting the importance of understanding programming as a cultural practice that involves communication, collaboration, and creativity. By viewing programming through the lens of literacy, we can recognize its importance as a means of expression and cultural production, while also working to promote greater equity and accessibility in the digital world.

One quote that particularly stood out to me was from Blackwell et al. (2022), who write that "live coding is less about producing correct programs than about discovering new musical experiences through programming." This idea challenges traditional notions of programming as a process of creating fixed and functional systems, and instead emphasizes the importance of experimentation and exploration in programming. It highlights the potential for programming to be used as a means of creative expression and discovery, rather than simply as a technical tool.