

Introduction to Computer Science
Homework Packet
March 23rd – March 27th

Students, please complete this packet on days that we would normally have class (Tuesday and Friday). Although school has been cancelled, I want to ensure that you are still getting the necessary work to keep your minds fresh and ready for when we return back to school.

Day #1

Quick Start: Please answer the following and explain your reasoning.

“Is excessive smartphone use affecting the mental health of teens?”

Activity: Complete Worksheet #1

“Early Programming: Crash Course Computer Science #10”

Directions: Watch the video (link is on the handout) and answer the questions

Day #2

Quick Start: Please answer the following and explain your reasoning.

“What are the structural and social barriers students face at home, in schools, and in society that could influence their likelihood to enter the computer science field?”

Activity: Complete Worksheet #2

“First Programming: Crash Course Computer Science #11”

Directions: Watch the video (link is on the handout) and answer the questions

BONUS ACTIVITY: Making a Peanut Butter and Jelly Sandwich

Directions: You will attempt to “program” someone to make a peanut butter and jelly sandwich.

1. To begin, please write down instructions for making a peanut butter and jelly sandwich.
2. Once you are done writing your instructions, give them to someone and inform them that they are a computer and that they are going to follow your exact instructions.
3. Evaluate your ability to give adequate instructions (i.e. what went good, bad, etc..)

Early Programming: Crash Course Computer Science #10

Available at <https://youtu.be/rtAIC5J1U40> or just youtube/google "Crash Course Computer Science 10"

1. How did **Joseph Marie Jacquard**'s programmable **textile loom** function in 1801?



- a. Many consider Jacquard's loom to be one of the earliest forms of _____.
2. The world's first General-Purpose electronic computer, the _____, completed in 1946, used a ton of plugboards.
 3. By the late 1940s and into the 50s, _____ was becoming feasible. Instead of storing a program as a physical plug board of wires, it became possible to _____ entirely in a computer's memory, where it could be easily changed by programmers and quickly accessed by the CPU. These machines were called _____.
 4. What are the hallmarks of a **von Neumann computer**?
 5. The largest program ever punched into punch cards was the US Air Force's _____ completed in _____.
 6. A close cousin to punch cards was _____, which is basically the same idea, but continuously instead of being on individual cards.
 7. What was **panel programming**?

8. The first commercially successful home computer was the _____, which was sold in two versions: _____ and as a _____.
9. Programming these early computers was the realm of _____: either _____ who did this for a living or _____.



First Programming Languages: Crash Course Computer Science #11

Available at <https://youtu.be/RU1u-js7db8> or just youtube/google "Crash Course Computer Science 11"

10. Computer **hardware** can only handle raw, _____ instructions. This is the "_____" computer processors natively speak. It's called **Machine Language** or _____.

11. An informal, high-level description of a program is called _____.

12. What is an **Assembler**? What function does it perform?

13. **Dr. Grace Hopper** designed a high-level programming language called "_____" _____, or **A-0** for short.

a. A **compiler** is a specialized program that _____ "source" code written in a programming language into a low-level language, like assembly or the binary "machine code" that the CPU can directly process.

14. _____, derived from "Formula Translation", was released in 1957 and came to dominate early computer programming.

15. Computer experts from industry, academia, and government formed a consortium in 1959 to guide the development of a _____ that could be used across different machines. The result was the high-level, easy to use, **Common Business-Oriented Language**, or _____.

16. Thanks to these languages, computing went from a cumbersome and esoteric discipline to a _____ and _____ tool.

17. What's one or two programming language(s) that developed in each of these decades?

1960s:

1970s:

1980s:

1990s:

2000s:

18. Each new language attempts to leverage new and clever _____ to make some aspect of programming easier or more powerful or take advantage of emerging technologies and platforms.

