

Layers of a Computer exercise

Materials:

Construction paper: red, yellow, green, purple

Glue or glue sticks

Layers of a Computer and Abstraction slides (see the deck in repo)

Before class: have the construction paper cut into any shape.

As the preceding layers are given to the students, have them glue the layers to one another using the outline that you drew on the layers.

- Hand each student the unlabeled -
 - purple *information* layer paper.
Explain that this is the core of the computer layer diagram and what this layer comprises of. The information in a computer is managed using binary digits: 0's and 1's. Without this data, hardware & software would be useless!
 - green *hardware* layer
Review that this is the physical stuff of the computer, transistors , circuits, vacuum tubes, keyboards, monitor, etc. Recall with the class that we organized the advancements of computer components into generations.
 - red *software* layer
Explain that this layer is made up of programming, operating systems, applications, etc. Programming, the instructions used to accomplish computations and manage data. The goal is to solve problems. The operating system, helps to manage a computer's resources. Applications Focuses on using the computer to solve specific real-world problems. This helps us take advantage of the computer's abilities, such as helping us to design a building or play/make a game.
 - Yellow *communication* layer
Communication is essential to how computer systems operate, which is why it encompasses all the layers. We can communicate with the computer to have it perform its programmed functions for us. We can use computer technology to communicate with the world.

While the class finishes glueing their layers together, explain that when we are dealing with one computer layer, we don't need to be thinking about the other layers, or how the other layers work. This notion is *abstraction*. Continue with the [slides](#) to further explain abstraction.

To abstract, means to pull something away from somewhere or from view.

Certain characteristics are hidden, so that we can see the necessary details. An abstraction leaves only the information necessary to complete our goal.

For example, when someone is driving a car, all they need to be focused on is actually driving the car and all the technical details are ignored. They may know those technical details, like how the engine works to move the car, but they only need to be focused on driving the car and paying attention to the road they are traveling on.

Again, when we are dealing with one computer layer, we don't need to be thinking about the other layers. For example, when we are writing a program, we don't need to concern ourselves with how the hardware carries out the instructions.

Abstraction is the key to computing!