Name:	Date:

Computer, Programs, Languages

What is a *computer*?

Describe in a few words how you would define a "computer":

Are *these persons* "computers"?

Hidden Figures
https://www.youtube.com/watch?v=5wfrDhgUMGI

Katherine Johnson
1918-2020

Mary Jackson
1921-2005

Dorothy Vaughan
1910-2008

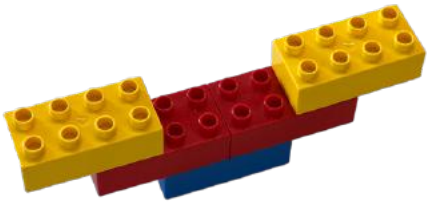
Are *you* a computer?

Perform the following addition:

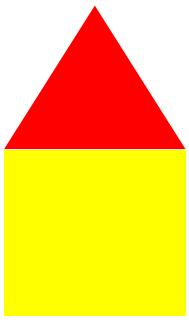
987654321 +1234567890 _____	Describe what you did:

What does it mean to *program*?

Define in a few words what you think “programming” means:



Describe the shown figure:



Describe the shown figure:

Natural language vs. *programming language*

	Natural language	**Programming language**
Definition	*Nel mezzo del cammin di nostra vita* *mi ritrovai per una selva oscura* *ché la diritta via era smarrita.*  Photo by Natalia Y. on Unsplash	**Syntax** t ::= terms: x variable λx.t abstraction t t application v ::= values: λx.t abstraction value **Evaluation** $\frac{t_1 \to t'_1}{t_1\ t_2 \to t'_1\ t_2}$ (E-App1) $\frac{t_2 \to t'_2}{v_1\ t_2 \to v_1\ t'_2}$ (E-App2) $(\lambda x.t_{12})\ v_2 \to [x \mapsto v2]t_{12}$ (E-AppAbs)
Used for…		
Examples… (that I know at least a little)		
Example texts…	Parlo italiano. _____ Deutsch. _____ français. I speak English.	print("Hello Python") System.out.println("Hello Java"); console.log('Hello JavaScript'); putStrLn "Hello Haskell"

What is the difference between natural languages and *programming languages*?

A programming language needs to be ***unambiguous***,
it **must not lead to misunderstandings**.
(It must be clear, unmistakable, and explicit.)

(KAPPLA activity)

Fundamental concepts of programming languages

All programming languages are defined by a:

SYNTAX	SEMANTICS
Syntax t ::= *terms:* x *variable* λx.t *abstraction* t t *application* v ::= *values:* λx.t *abstraction value*	**Evaluation** $\frac{t_1 \to t'_1}{t_1\ t_2 \to t'_1\ t_2}$ *(E-App1)* $\frac{t_2 \to t'_2}{v_1\ t_2 \to v_1\ t'_2}$ *(E-App2)* $(\lambda x.t_{12})\ v_2 \to [x \mapsto v2]t_{12}$ *(E-AppAbs)*

Programming languages are defined with mathematical precision. There is no room for interpretation. The **meaning of everything** is 100% clear.

Thanks to this **uniqueness**, the computer is able to automatically detect if the program violates any language rules. **It can help the programmer** to fix the problem by providing an error message.

Program with error	Kind of error	Correct program
rectangle(100, 200, yellow	**Syntax** Error	rectangle(100, 200, yellow)
rectangle("Ciao", 200, red)	**Semantic** Error	rectangle(100, 200, red)

While the computer can detect when a program **violates the rules of the language, it** can't detect whether the program will do what the programmer actually wanted to do. **It can't see in the programmer's head!**

Program with error	Kind of error	Correct program
rectangle(200, 200, yellow)	**Logic** Error	rectangle(100, 200, yellow)

Complete the following summary table:

Your computer **can** detect:	The computer **cannot** detect:

The Python programming language

This is a program written in Python:	Act like a computer and execute the program (draw here what it produces):
`rectangle(200, 100, black)`	

Inventor of Python:

	***Guido Van Rossum:*** A Dutch computer scientist Python was born by chance in the early 90s, during a moment of leisure and pastime. By 2023 about half (49%) of software developers worldwide used Python! Photo by Kushal Das on Wikimedia Commons. CC BY-SA 4.0

The name “Python” was inspired by (select one):

Photo by David Clode on Unsplash

○

Monty Python’s Flying Circus

○

Logo of Python:

Official definition of Python:

https://www.python.org/