

What is programming?

Programming = giving a computer instructions to achieve a specific goal

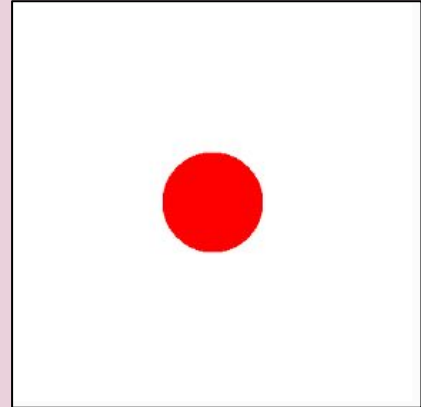
The problem is broken down into singular steps which are executed sequentially. This sequence is called an **algorithm**.

Goal: Draw a red circle in the center of the screen.

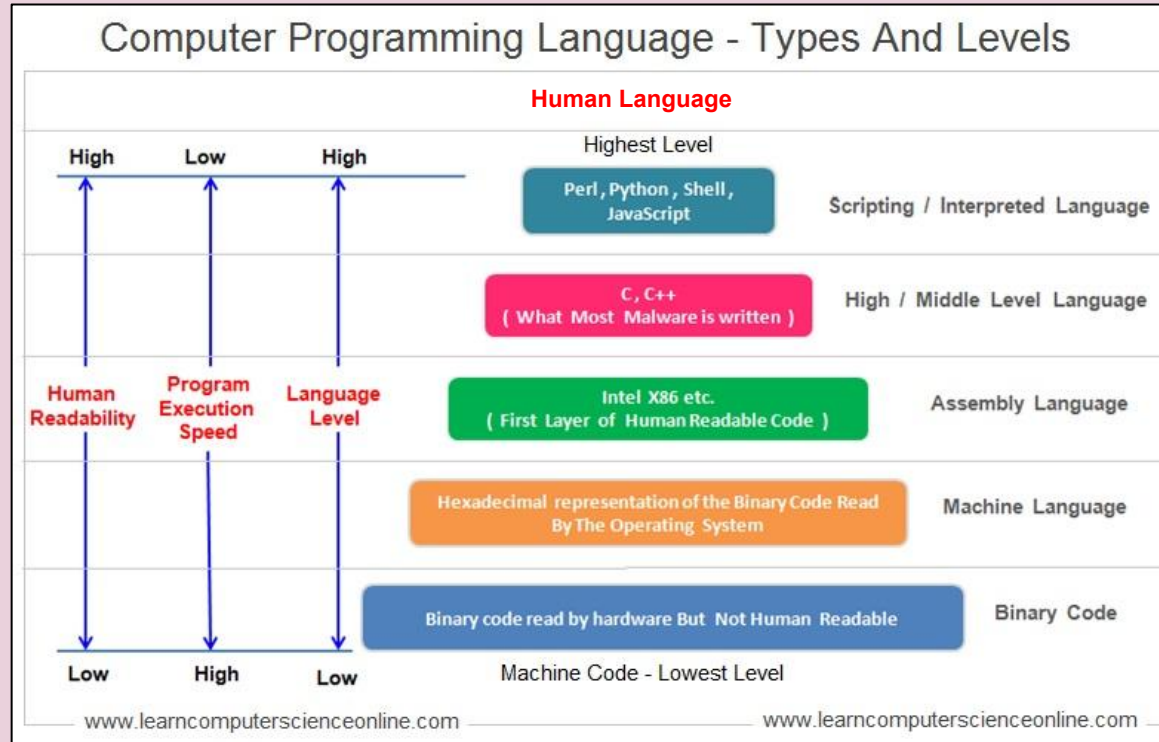
How does the computer know that these 3 lines of code should draw a red circle?
(a computer doesn't know the words "circle" or "fill")

→ In using a language the computer understands: a **programming language**

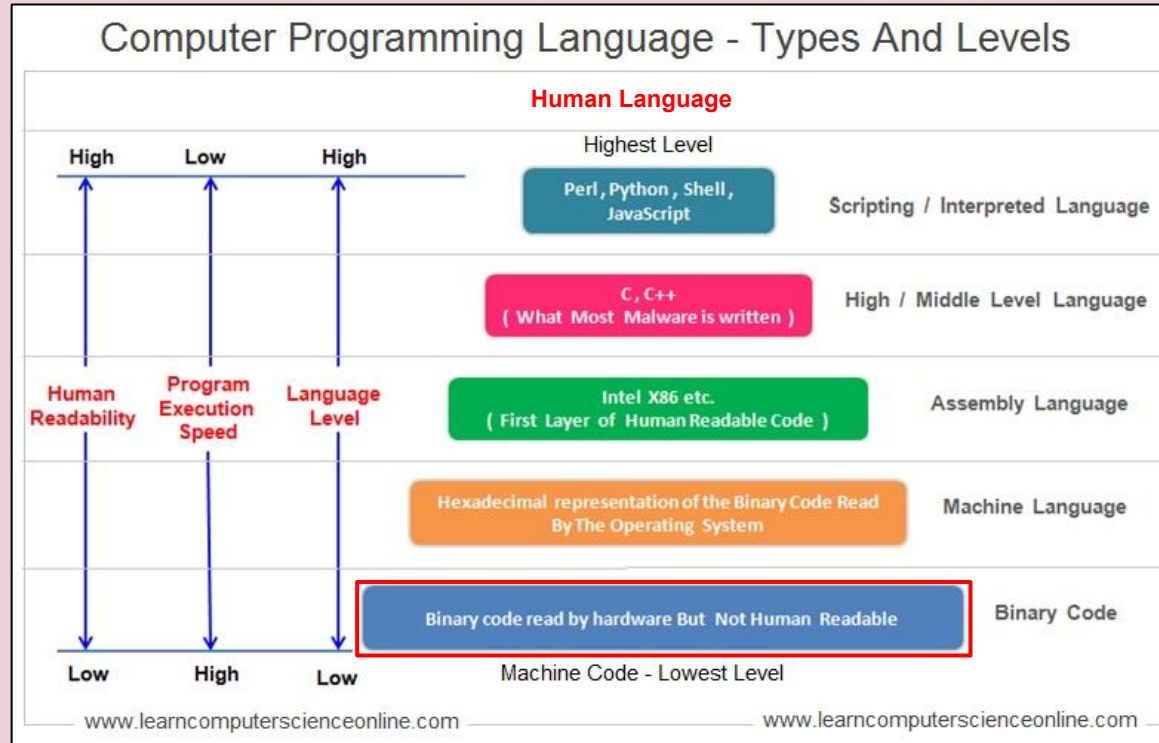
```
void draw() {  
  fill(255, 0, 0);  
  noStroke();  
  ellipse(width/2, height/2, 100, 100);  
}
```



Programming languages serve as a **layer of abstraction** between the hardware and a human. Different programming languages are thereby closer to the hardware (less abstracted) or human language (highly abstracted).



Programming languages serve as a **layer of abstraction** between the hardware and a human. Different programming languages are thereby closer to the hardware (less abstracted) or human language (highly abstracted).

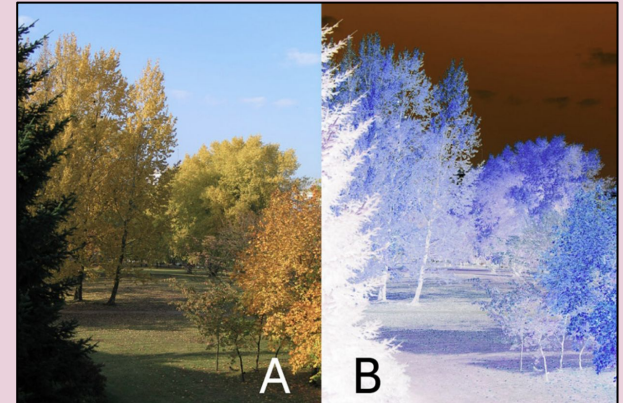
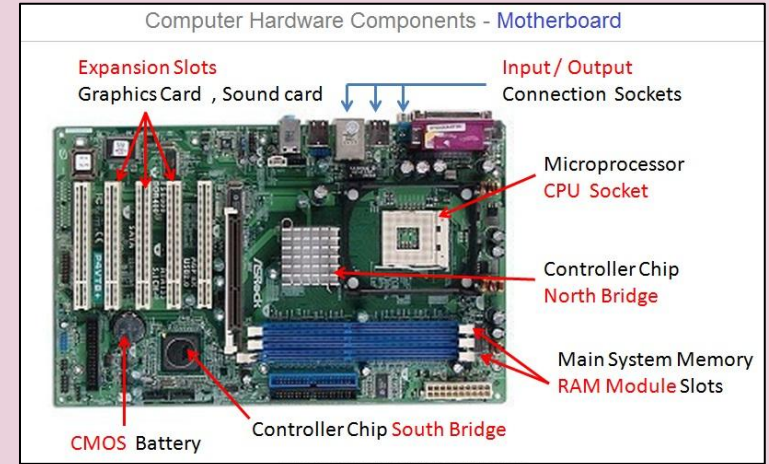


Lowest level (Hardware):

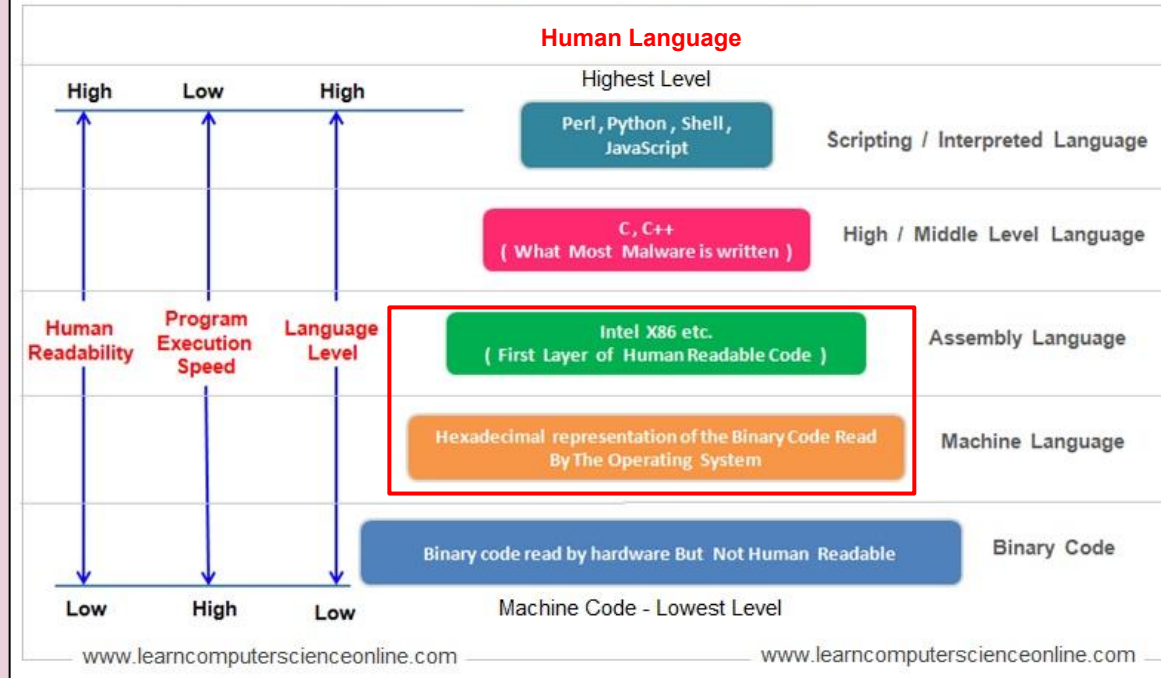
Different components like CPU, memory etc. communicate via voltages. This communication can be represented as **binary code**. Existing voltage is encoded as 1, no voltage as 0. The whole sequence of component-interactions is a **program**.

Real world **data** can be binary encoded. By using data inside programs, it can be manipulated and finally stored back into **memory**.

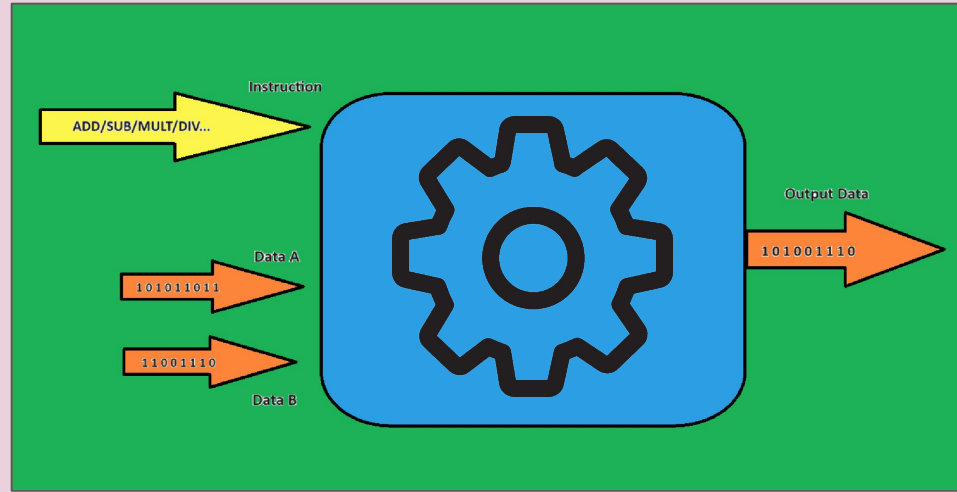
Computer = Data storage and data manipulator



Computer Programming Language - Types And Levels



Normal instructions			Compressed	
00000	SUB	10000	CMP	000 SUB
00001	AND	10001	TEST	001 AND
00010	ADD	10010	LW	010 ADD
00011	OR	10011	SW	011 CMP
00100	XOR	10100	LH	100 LW
00101	LSR	10101	SH	101 SW
00110	LSL	10110	LB	110 LDI
00111	ASR	10111	SB	MOV
01000	BREV	11000		
01001	LDILO	11001	LDI	
01010	MPYUHI			Reserved for FPU
01011	MPYSHI	Special Insn		
01100	MPY	11100	BREAK	
01101	MOV	11101	LOCK	
01110	DIVU	11110	SIM	
01111	DIVS	11111	NOOP	

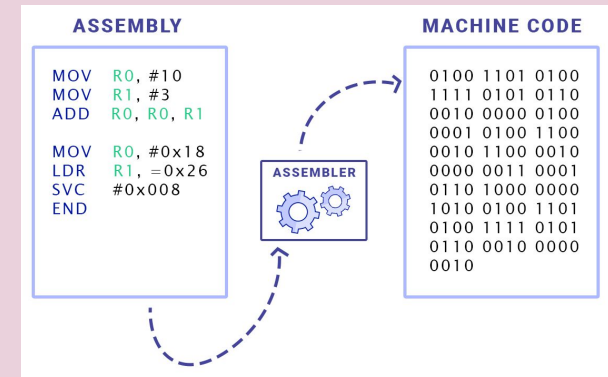


Second level (machine code):

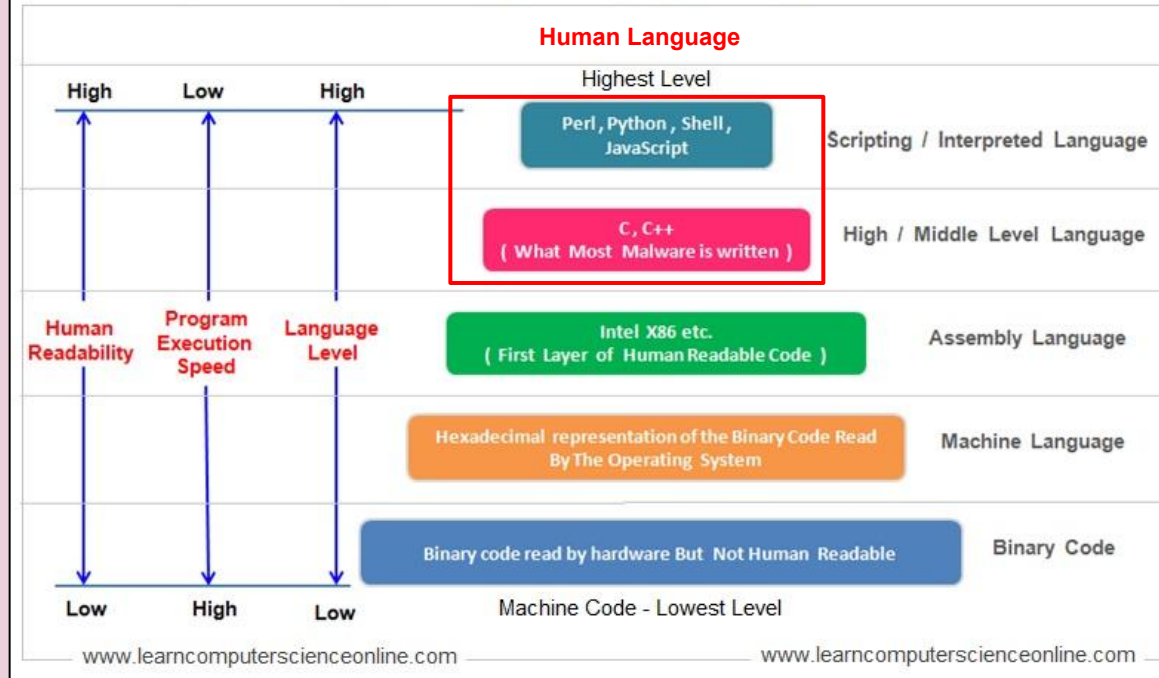
Which operations are executed on the data is controlled via different control commands.

Third level (Assembly):

Human-readable representation of machine code.



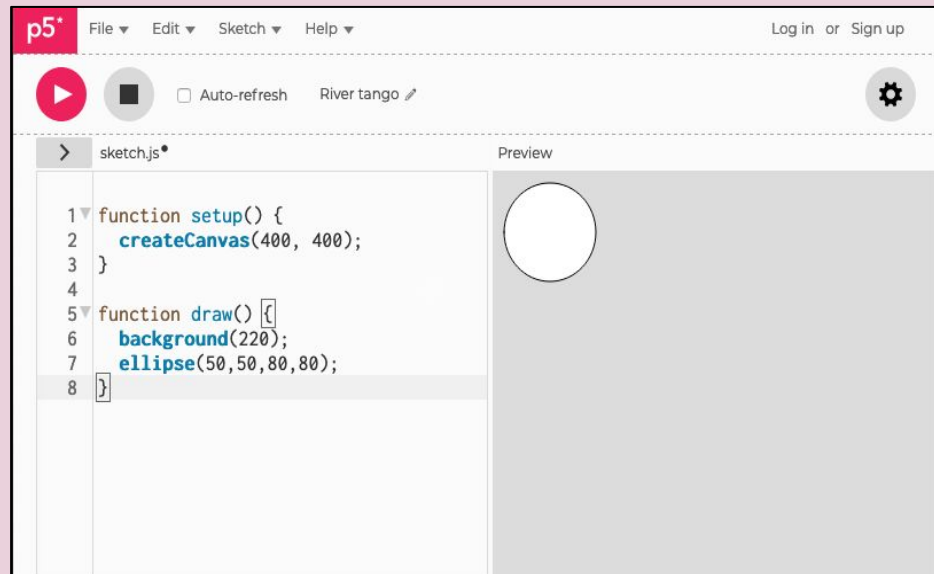
Computer Programming Language - Types And Levels



```

#include <stdio.h>
#include <stdlib.h>
int main(){
    int *ptr =NULL;
    int n =5;
    ptr = (int*)malloc(sizeof(int)*5); //memory allocated for 20 bytes
    if(ptr == NULL)
    {
        printf("Failed to allocated");
    }
    else
    {
        for(int i=0;i<n;i++)
        {
            *(ptr + i) = i;
        }
    }
    for(int i=0;i<n;i++)
    {
        printf("%d ",*ptr+i);
    }
    free(ptr);
    return 0;
}

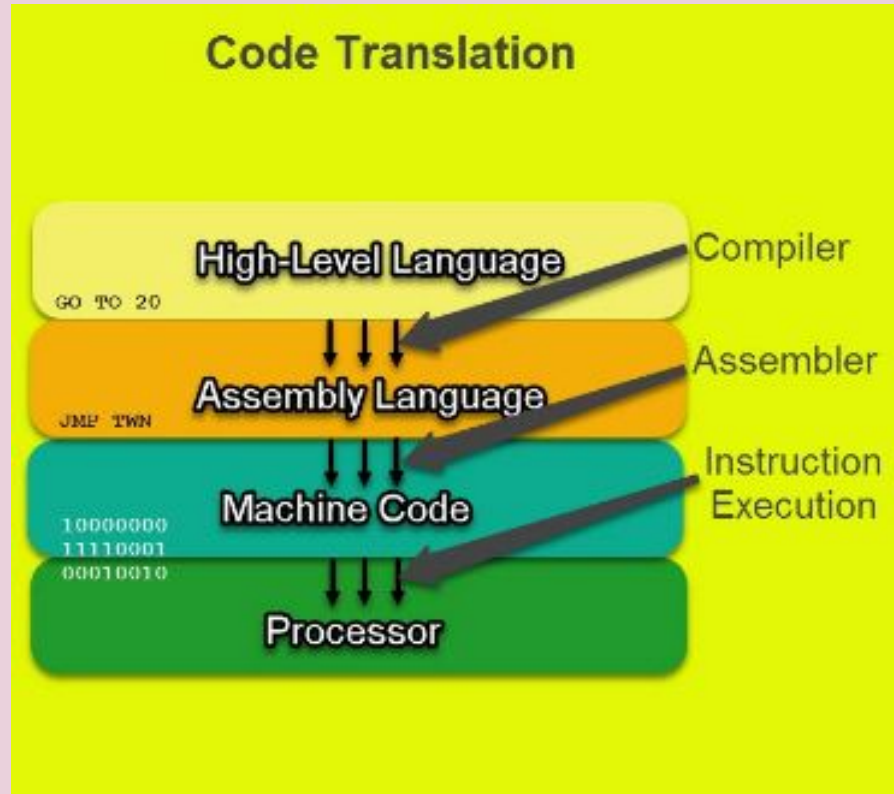
```



Fourth/Fifth level (high-level languages):

“Easily” readable. Code highly abstracted from the hardware.

In using different steps, the code is translated back to the hardware level.



*"p5.js is a JavaScript **library** for creative coding, with a focus on making coding accessible and inclusive for artists, designers, educators, beginners, and anyone else!"*

- highly abstracted, therefore easy to learn
- interactive learning experience as code directly produces visible results

```
function setup() {  
  createCanvas(400, 400);  
}  
  
function draw() {  
  background(255);  
  fill(255, 0, 0);  
  circle(width / 2, height / 2, 100);  
}
```

