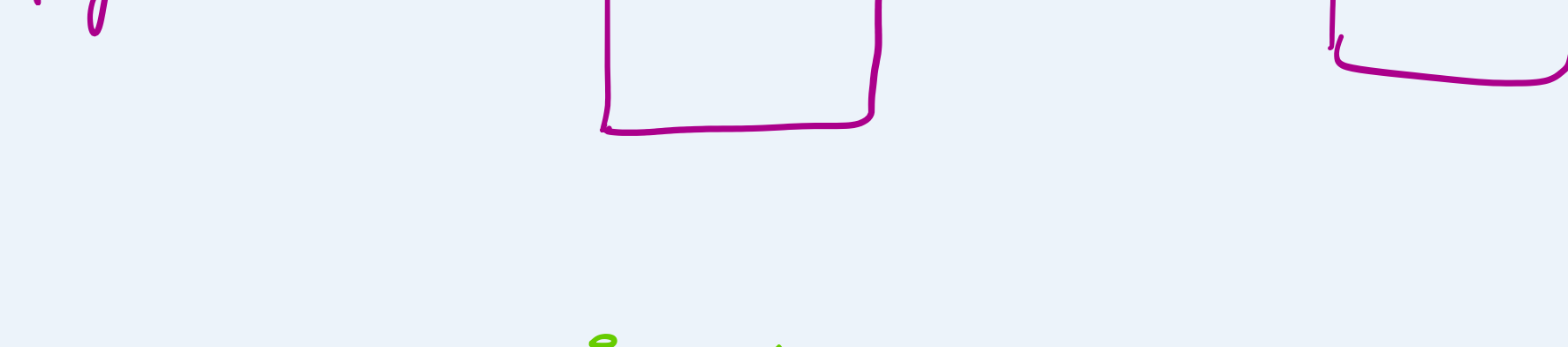
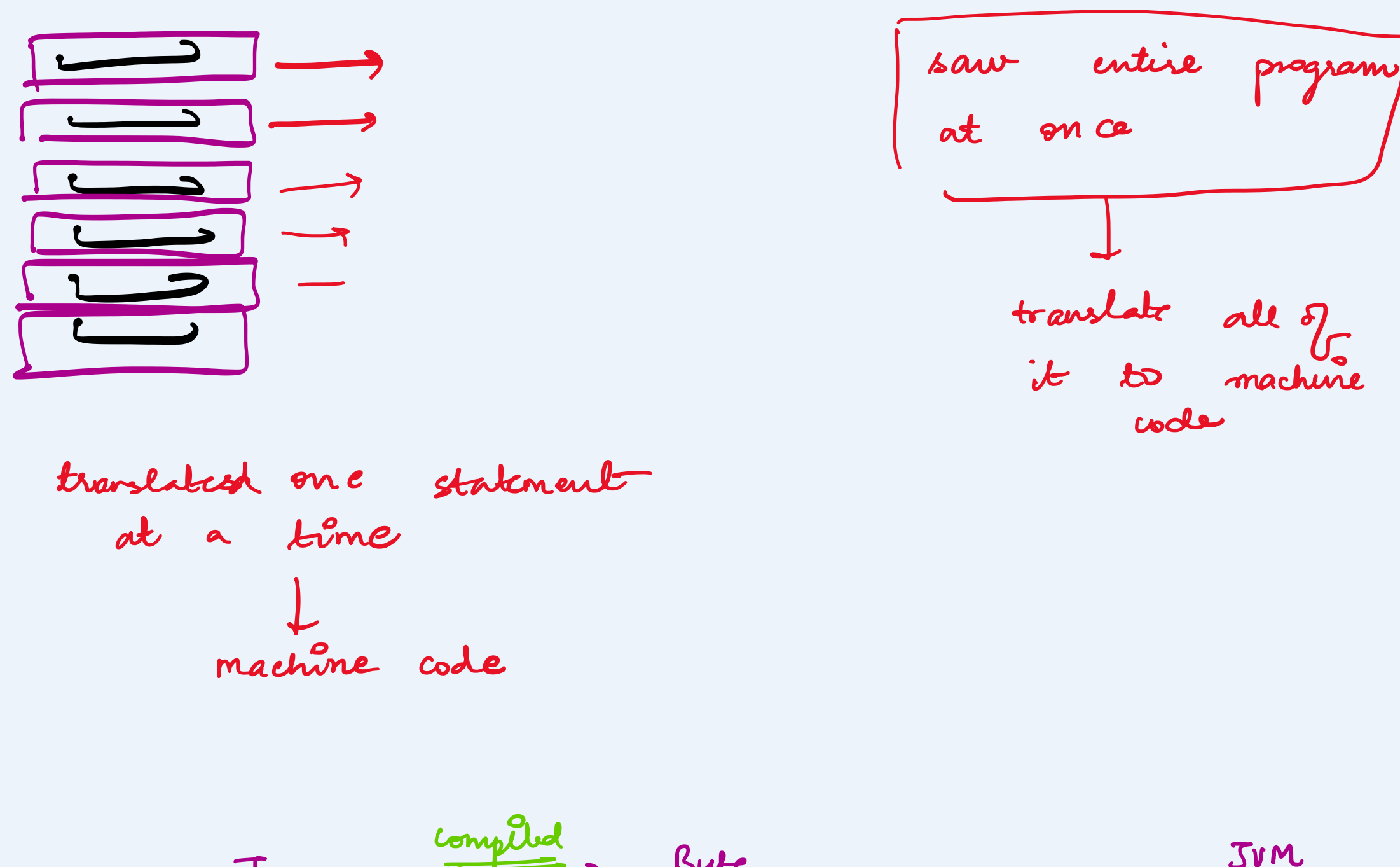
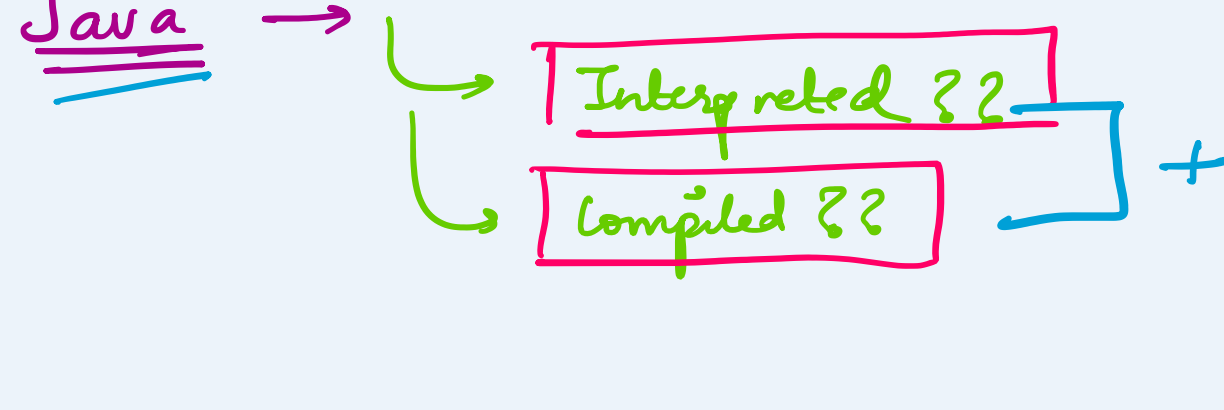


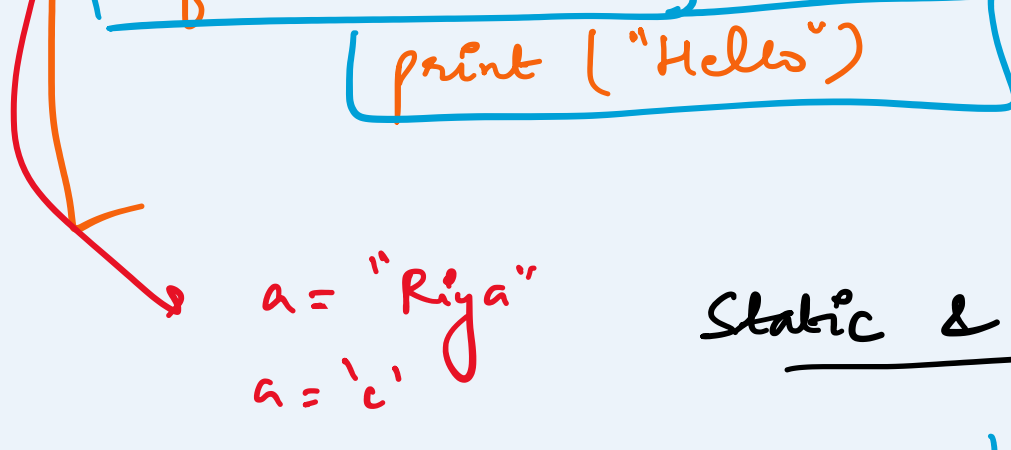
## Transition to C++

① C++ → Bjarne Stroustrup, 1980s



C++ → compiled language

Python → interpreted language



a = "Riya"  
a = 'c'

### Static & Dynamic typing

Java, C++

int val = 10 xx

char val

string val

(statically typed languages)

Python

a = 10  
a = "Akhil"  
a = 'c'

- |   |   |
|---|---|
| <p>① <u>Comments</u></p> <p style="color: red;">Java //</p> <p style="color: red;">C++ /* */</p> <p>② <u>Classes</u></p> <p style="color: red;">C++ → <span style="color: red;">int</span> <span style="border: 1px solid red; padding: 2px;">main()</span> <span style="color: red;">i</span></p> <p>④ <u>Libraries</u></p> <p style="color: red;">import</p> <p>⑤ <u>System.out.println("Hello")</u><br/><u>scn.nextInt()</u></p> | <p style="color: red;">C++</p> <p style="color: red;">//</p> <p style="color: red;">/* */</p> <p style="color: red;">Hello</p> <p style="color: red;">int <span style="border: 1px solid red; padding: 2px;">main()</span> i</p> <p style="color: red;">import</p> <p style="color: red;">cout &lt;&lt; "Hello" &lt;&lt; endl</p> |
|---|---|

## Data Types and operators

- ① short
- ② int
- ③ long
- ④ float
- ⑤ double
- ⑥ char

⑦ Booleans

bool  
0 → false  
non-zero → true

⑧ byte xx

Typecasting → same

== != < > >= <=

### operator overloading

② pointers

to provide the operators with a special meaning for a data-type

Strings → C++

### 2 types of strings

① char array → "Riya"  
single char

② String class → String val = "class"

Java String str = "Riya"

s.size()  
s.length()  
s.empty()  
s.find()  
s.substr(0)

⊕  
"abc" + "def"  
↓  
abcdef

substring of length n  
starting at position i

C++ strings are mutable

s[index] → ✓  
s[index] = new char

s.insert(index, t)

"abc" → abcde

abcde

Java

final

C++

const

long  
double  
Big Integer

long long  
long int →  
long long int →  $(-2^{63})$  to  $(2^{63}) - 1$   
short int

class paper {  
public:  
[ ]  
private:  
[ ]

class paper {  
private:  
public:  
private:

### break, continue

### Arrays

