

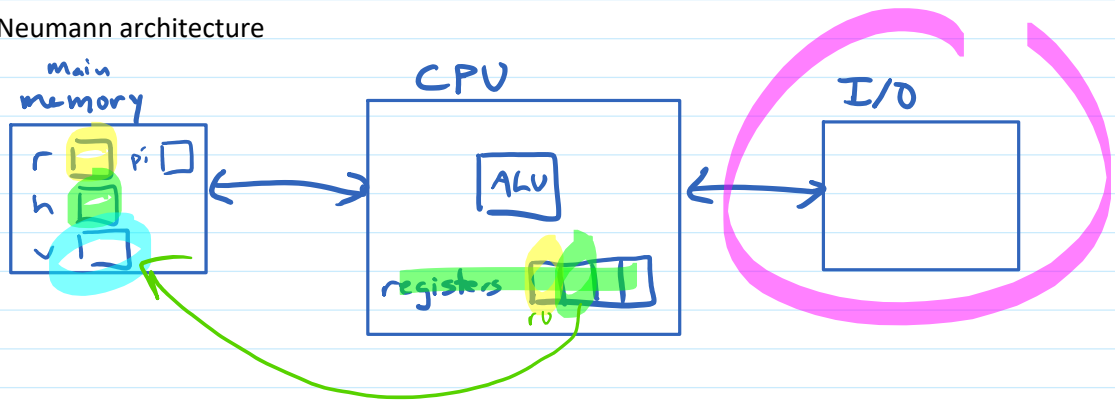
Source Code Translation



CPU: Qualcomm Snapdragon 845

CPU : Intel Core i5 6300U

von Neumann architecture



$$v = \pi * r * r * h$$

C compiler

C compiler

CPU 1 (maybe Snapdragon 845) Machine Instructions

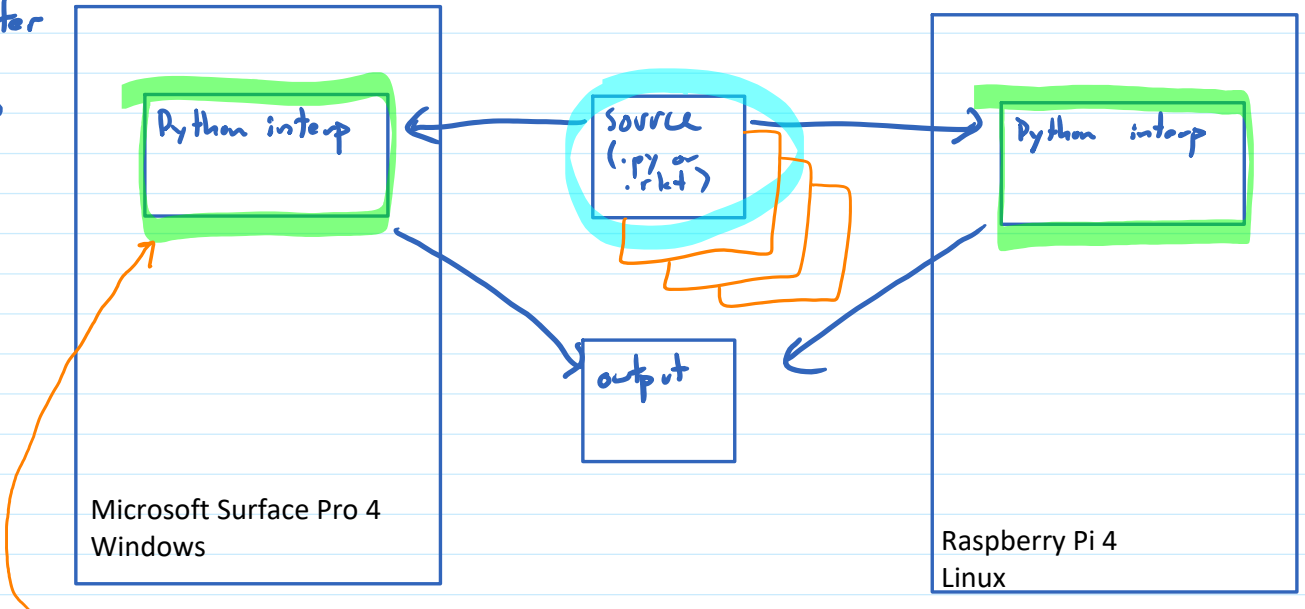
load r into register 1
 multiply r1 by r1, store result in r1 r^2 in R1
 load h into r2
 multiply r1 by r2, store result in r2 r^2h in R2
 load pi into r3
 multiple r2 by r3, store result in r3
 store r3 in v

load r into r1
 square r1
 mult r1 by h r^2h in r1
 mult-pi r1
 store r1 in v

Interpreters vs Compilers

Interpreter

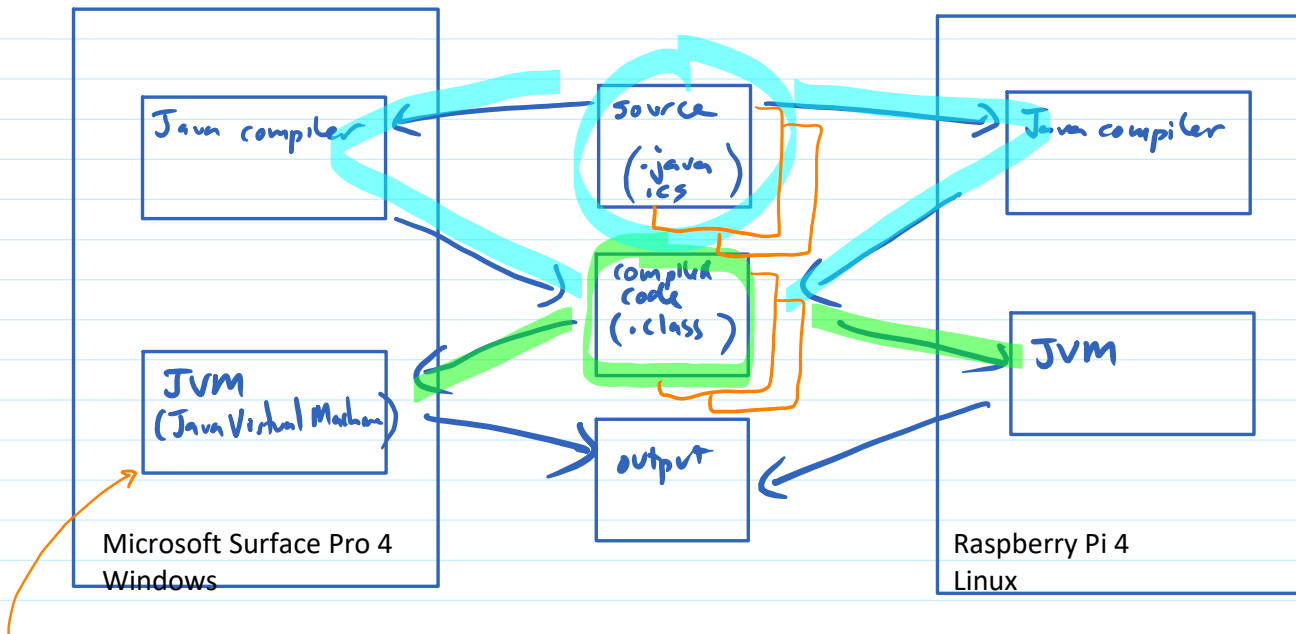
Python
Racket



translation from high-level source code to CPU instructions happens here (likely not explicitly [as in by producing a list of instructions] but implicitly by choosing to run different parts of the interpreter, which is itself a list of machine instructions)

Java
C#

(compiled
interpreted)



like the interpreter above, except the input is machine instructions for the virtual machine inside the .class files instead of high-level source code

