

# Virtual machine: lecture plan

---

## Overview

✓ The road ahead

➡ • Program compilation

## VM abstraction

- the stack
- memory segments

## VM implementation

- the stack
- memory segments

## VM implementation platforms

- VM emulator
- VM translator

## The VM translator

- Proposed implementation
- Building it (project 7)

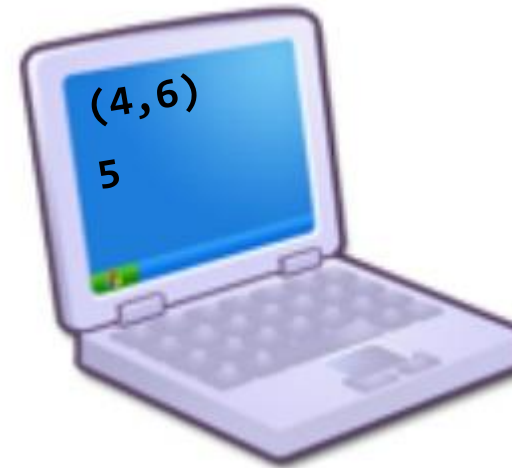
# The big picture

---

## High-level program

```
/** Demo: working with Point objects */
class Main {
  function void main() {
    var Point p1, p2, p3;
    let p1 = Point.new(1,2);
    let p2 = Point.new(3,4);
    let p3 = p1.plus(p2);
    do p3.print(); // prints (4,6)
    do Output.println();
    do Output.printInt(p1.distance(p3));
    return;
  }
}
```

```
/** Represents a Point */
class Point {
  field int x, y;
  static int pointCount;
  /** Constructs a new point */
  constructor Point new(int ax, int ay) {
    let x = ax;
    let y = ay;
    let pointCount = pointCount + 1;
    return this;
  }
  ...
}
```



# The big picture

## High-level program

```
/** Demo: working with Point objects */
class Main {
  function void main() {
    var Point p1, p2, p3;
    let p1 = Point.new(1,2);
    let p2 = Point.new(3,4);
    let p3 = p1.plus(p2);
    do p3.print(); // prints (4,6)
    do Output.println();
    do Output.printInt(p1.distance(p3));
    return;
  }
}
```

```
/** Represents a Point */
class Point {
  field int x, y;
  static int pointCount;
  /** Constructs a new point */
  constructor Point new(int ax, int ay) {
    let x = ax;
    let y = ay;
    let pointCount = pointCount + 1;
    return this;
  }
  ...
}
```



Compiler

## Low-level code

```
0000000000010000
1110111111001000
0000000000010001
1110101010001000
0000000000010000
1111110000010000
0000000000000000
1111010011010000
0000000000010010
1110001100000001
0000000000010000
1111110000010000
0000000000010001
0000000000010000
1110111111001000
0000000000010001
1110101010001000
0000000000010000
1111110000010000
0000000000000000
1111010011010000
0000000000010010
1110001100000001
0000000000010000
1111110000010000
0000000000010001
1111110000010000
...
```

# Program compilation: 1-tier

---

high-level  
program



this  
computer



that  
computer

...



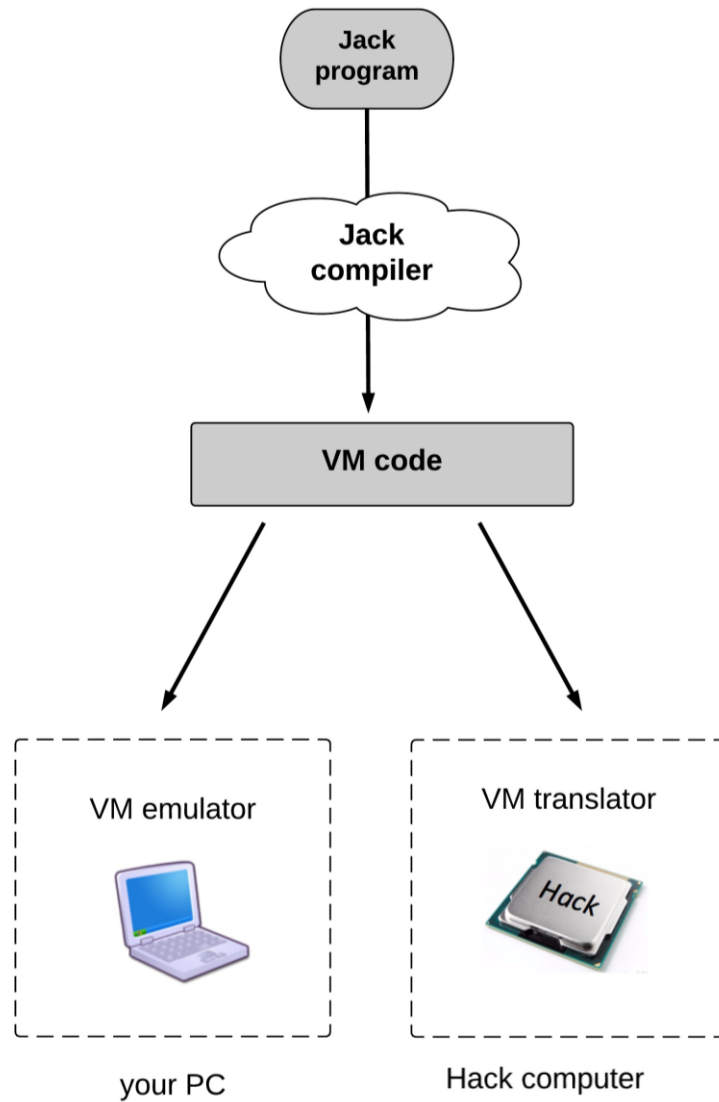
this  
device



that  
device

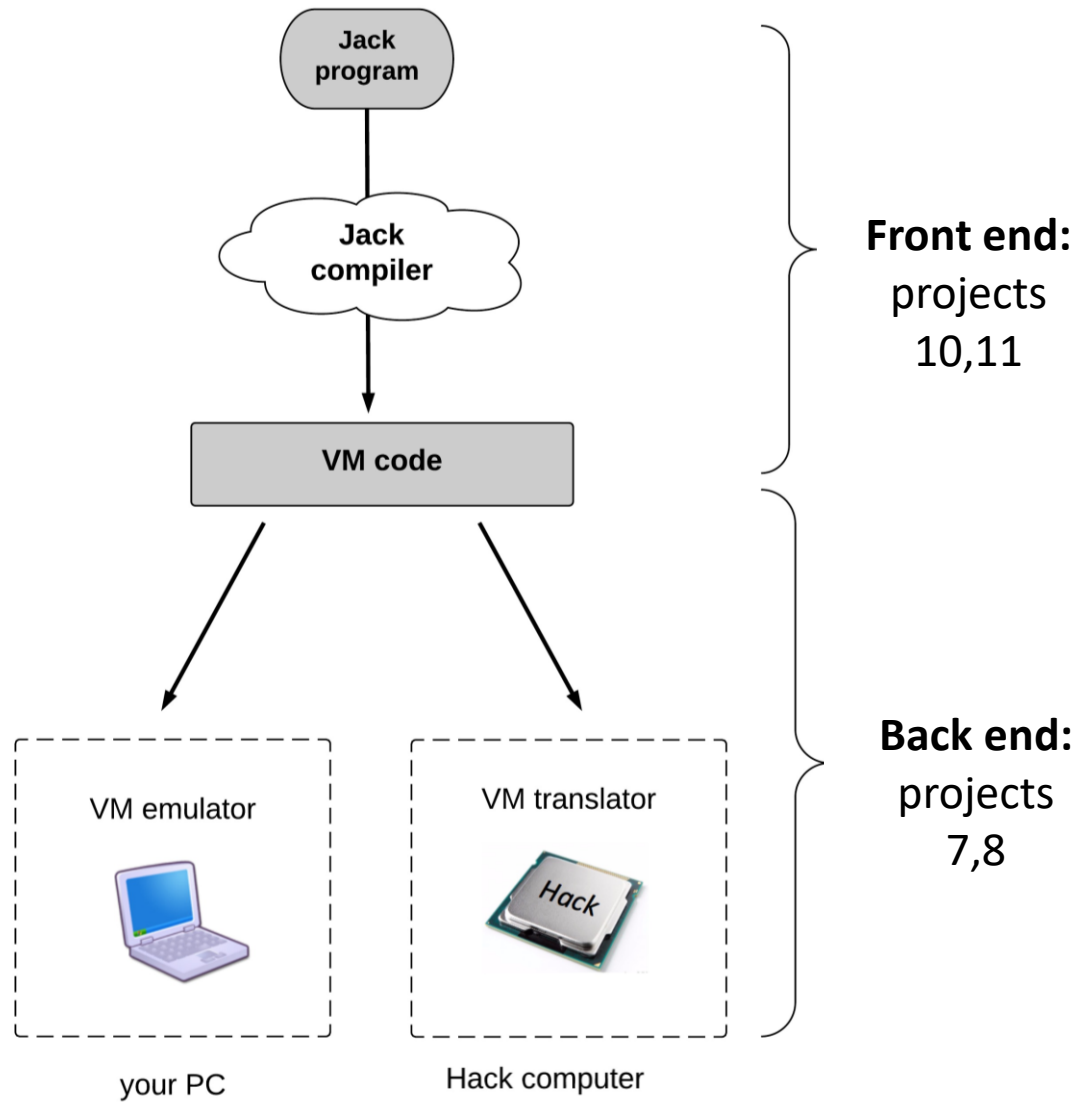
# Jack compilation

---

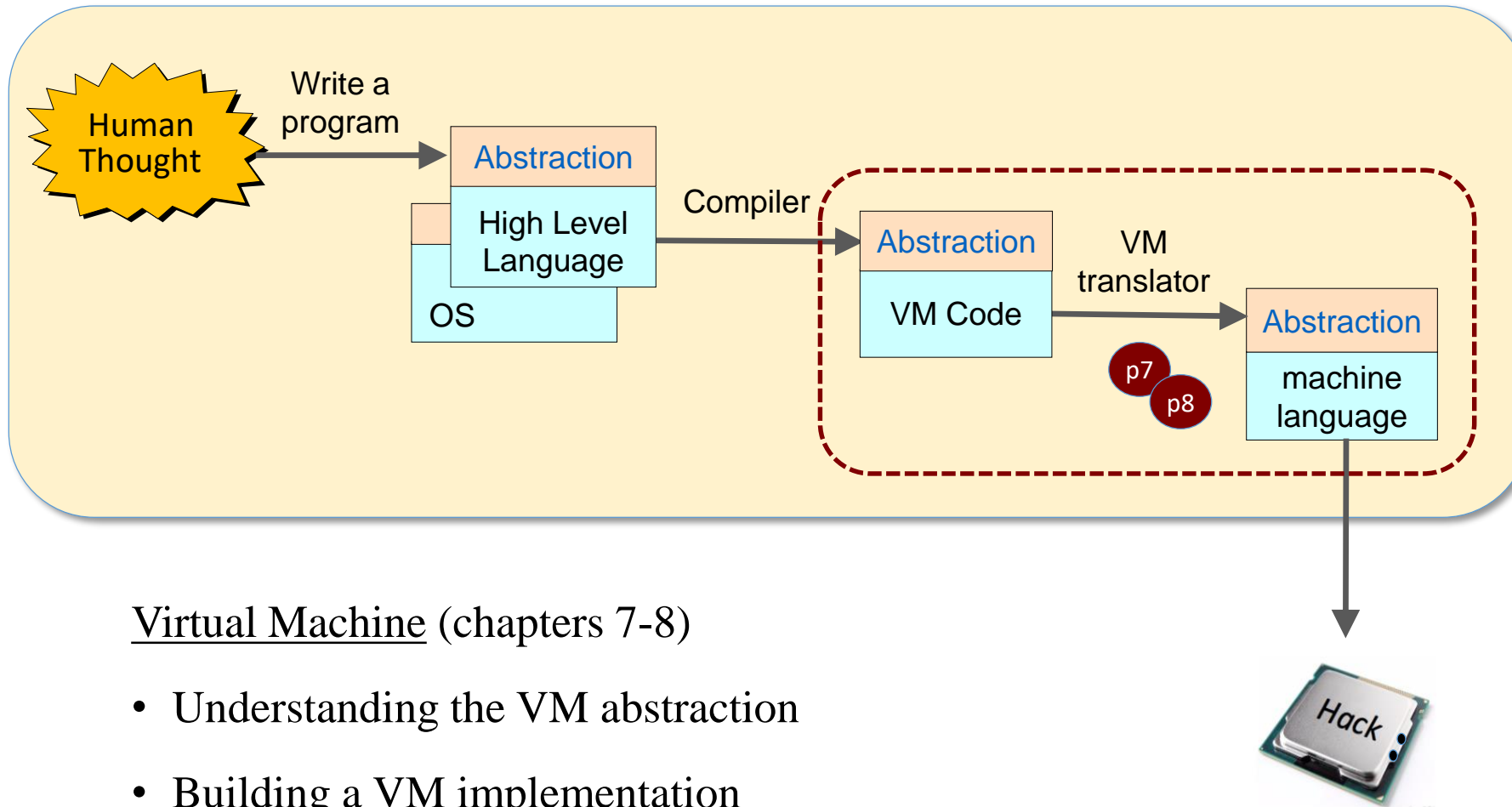


# Jack compilation

---



# The big picture



## Virtual Machine (chapters 7-8)

- Understanding the VM abstraction
- Building a VM implementation

# Take home lessons

---

- Compilation (big picture)
- Virtualization
- VM abstraction
- Stack processing
- VM implementation
- Pointers
- Programming.