Fast campus

CS SCHOOL

1. Virtual Stack Machine
   - stack-based

2. Operand Stack

How does VM work?

```
>>> def add(a, b):
        c = a + b
        return c

>>> import dis
>>> dis.dis(add)
  2           0 LOAD_FAST                0 (a)
              2 LOAD_FAST                1 (b)
              4 BINARY_ADD
              6 STORE_FAST               2 (c)

  3           8 LOAD_FAST                2 (c)
             10 RETURN_VALUE
```

disassembling

Fast campus

How does VM work?

```
 0 LOAD_FAST          0 (a)
 2 LOAD_FAST          1 (b)
 4 BINARY_ADD
 6 STORE_FAST         2 (c)

 8 LOAD_FAST          2 (c)
10 RETURN_VALUE
```

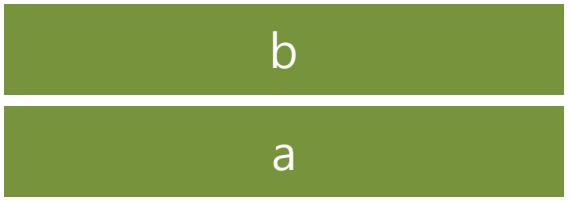
Operand stack

Fast campus

How does VM work?

```
 0 LOAD_FAST        0 (a)
 2 LOAD_FAST        1 (b)
 4 BINARY_ADD
 6 STORE_FAST       2 (c)

 8 LOAD_FAST        2 (c)
10 RETURN_VALUE
```

Operand stack



Fast campus

How does VM work?

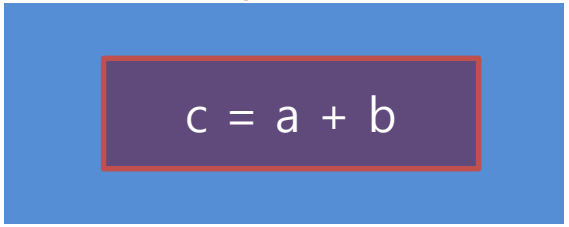
```
 0 LOAD_FAST                0 (a)
 2 LOAD_FAST                1 (b)
 4 BINARY_ADD
 6 STORE_FAST               2 (c)

 8 LOAD_FAST                2 (c)
10 RETURN_VALUE
```

Operand stack

a + b

How does VM work?

Private heap




```
 0 LOAD_FAST                0 (a)
 2 LOAD_FAST                1 (b)
 4 BINARY_ADD
 6 STORE_FAST               2 (c)

 8 LOAD_FAST                2 (c)
10 RETURN_VALUE
```

Fast campus

How does VM work?

```
 0 LOAD_FAST          0 (a)
 2 LOAD_FAST          1 (b)
 4 BINARY_ADD
 6 STORE_FAST         2 (c)

 8 LOAD_FAST          2 (c)
10 RETURN_VALUE
```

Operand stack

c

Fast campus

How does VM work?

```
 0 LOAD_FAST                0 (a)
 2 LOAD_FAST                1 (b)
 4 BINARY_ADD
 6 STORE_FAST               2 (c)

 8 LOAD_FAST                2 (c)
10 RETURN_VALUE
```

returns c