Interpreter Optimization: Superinstructions

- **Identify basic blocks**
  - Straight-line code
  - That ends with some control flow
    - Typically branch, jump, or call
    - Exceptions are control flow but they occur in high-level languages for many instructions so, these instructions typically do not end basic blocks
      - If they did, there wouldn’t be any instructions to work with/combine

- **For each basic block**
  - Make a dispatch body (superinstruction)
  - **Remove dispatch code** in between VM instructions within block
    - Increment VM program counter (PC)
    - Extract address from VM instruction, jump to address

```
iload_handler
iadd_handler
istore_handler
```

```
0 iload_0
1 istore_1
2 iload_0
3 bipush 10
5 if_icmple 17
```