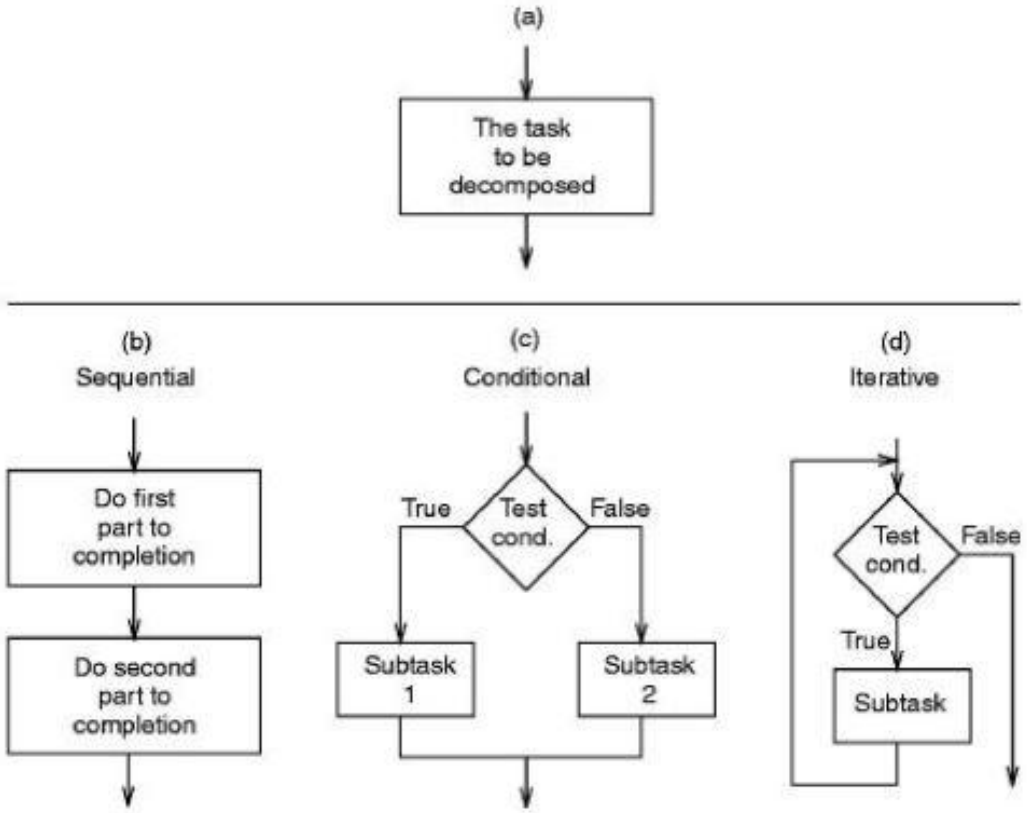
WEEK FOUR

Acknowledgements: Slides created based off material provided by Dr. Travis Doom

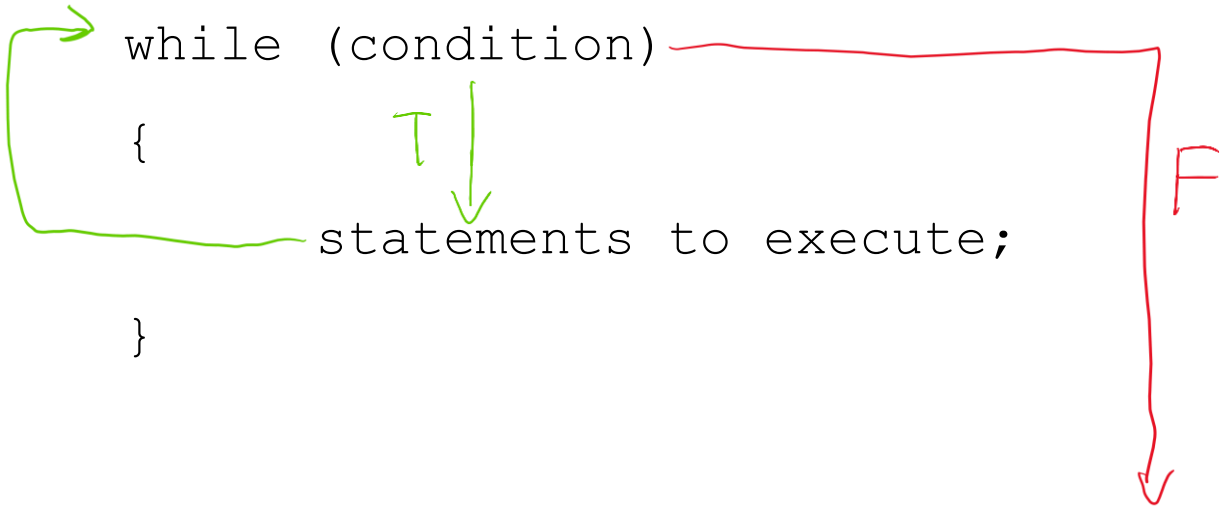
CONTROL STRUCTURES

- Sequential
  - Default
  - Do A -> B -> C -> …
- Selective/Conditional
  - Decision/choice
  - Do A if some condition, otherwise do B
- Iteration
  - Loops
  - Do A repeatedly until a condition is met



ITERATION: WHILE LOOPS

- Continues to execute a section of code while a condition is true



ITERATION: WHILE LOOPS

- While loops that never exit are possible (infinite loops)
- We want to avoid these

```
while (true)
{
    System.out.println("wee");
}
```

```
int counter = 5;

while (counter < 10)
{
    System.out.println(counter);
    counter = counter - 1;
}
```

counter = 5, 4, 3, 2...

ITERATION: WHILE LOOPS

```
int counter = 5;
while (counter > 0)
{
      System.out.println(counter);
      counter = counter - 1
}
```

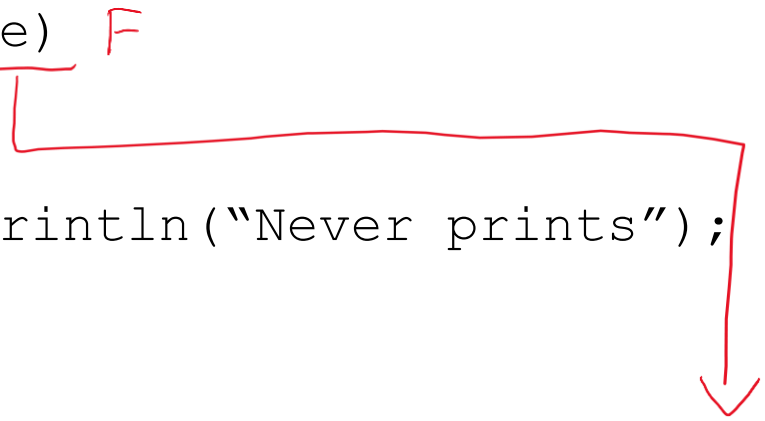
output:

5
4
3
2
1

ITERATION: WHILE LOOPS

- It is also possible for the entire loop to be skipped

```
boolean flag = false;
while (flag == true)
{
      System.out.println("Never prints");
}
```



ITERATION: WHILE LOOPS

- Off by one errors

```
// code to count up to 5
int num = 0;
while (num < 5)
{
        num = num + 1;
        System.out.println(num);
}
```

Output:

1
2
3
4
5

```
int num = 1;
while (num < 5)
{
        System.out.println(num);
        num = num + 1;
}
```

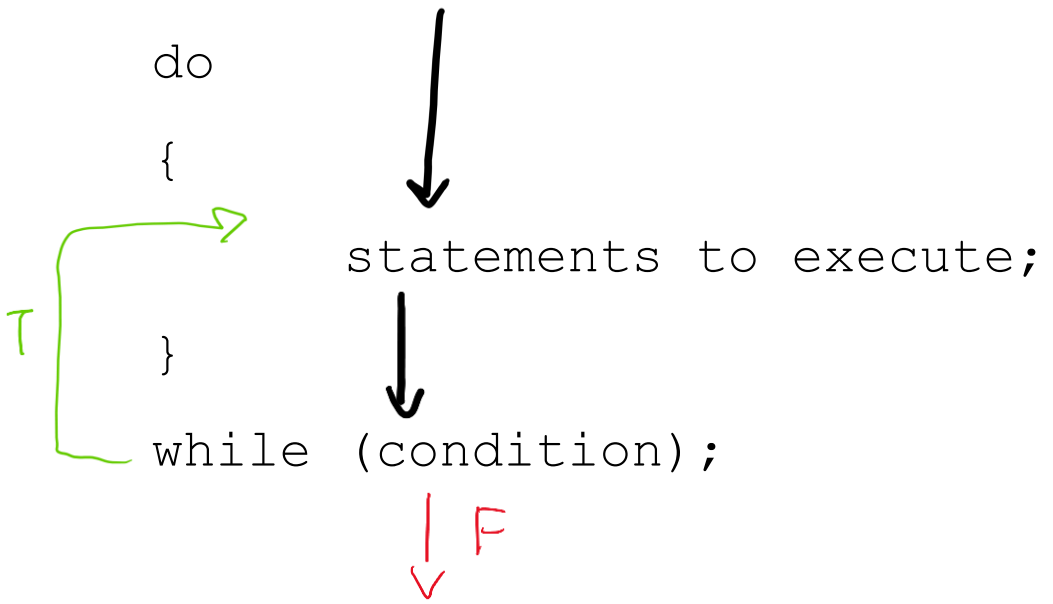
Output:

1
2
3
4

ITERATION: DO WHILE LOOPS

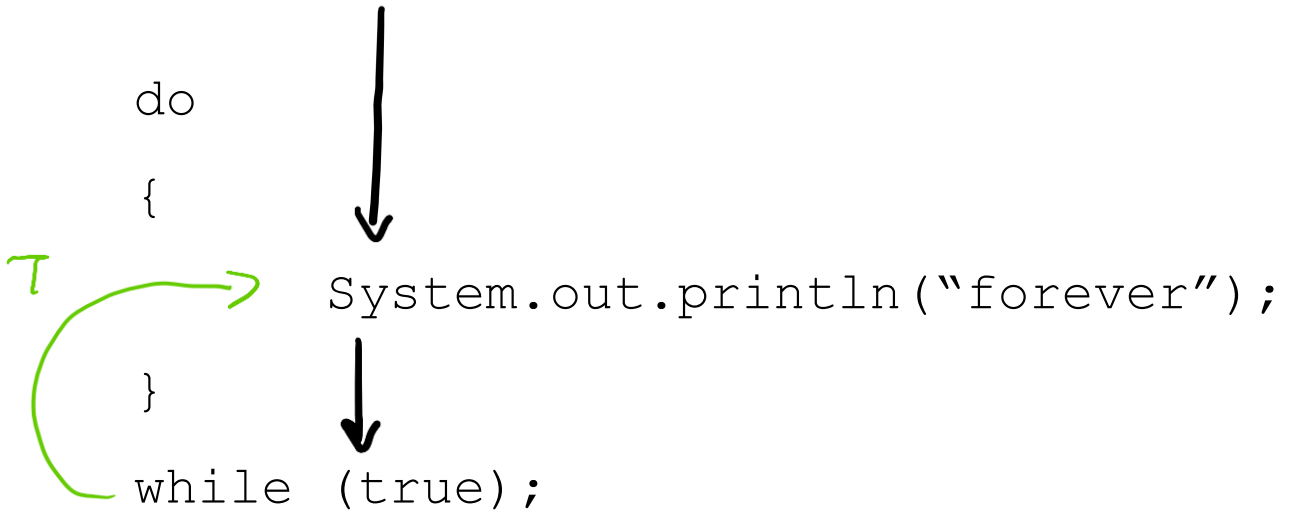
- Same as while loop but we execute our code before checking the condition

```
do
{
        statements to execute;
}
while (condition);
```



ITERATION: DO WHILE LOOPS

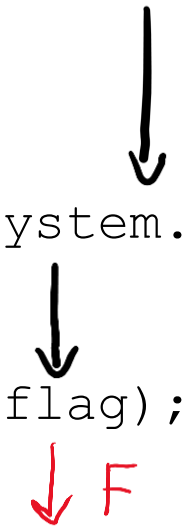
- Infinite loops are still possible



ITERATION: DO WHILE LOOPS

- It is not possible for the code in the do statement to be completely skipped

```
boolean flag = false;
do
{
      System.out.println("prints once but only once");
}
while (flag);
```



ITERATION: FOR LOOPS

- Loops with more power
- Count through iterations

```
for (initialization; condition; update)
{
      statements to execute;
}
```

ITERATION: FOR LOOPS

- Loops with more power
- Count through iterations

initialization conditional update

```
for (int i = 5; i >= 0; i--)
{
        System.out.println(i);
}
```

#1 Initialize integer i and assign the value 5 to it

#2 Check the condition
Is i >= 0?

#3 If true, print out i

#4 Update i by subtracting one

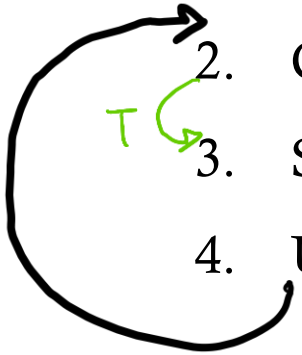
ITERATION: FOR LOOPS

```
for (int i = 5; i >= 0; i--){
        System.out.println(i);
}
```

init cond update

```
int i;
for (i = 5; i >=0; i = i - 1){
        System.out.println(i);
}
```

init assignment cond update



1. Initialization happens first
2. Conditional check happens next
3. Statements inside curly braces
4. Update executes last

IN CLASS ACTIVITY

- Countdown from 10
- For each number print “T-Minus” before the number
- Between 7 and 6, print “MAIN ENGINE START”
- After you get to 1
- Print out “LIFT OFF!!!”