loops also called "repetition statements"

- **The While Statement**
  ```
  while (condition) 
    statement;
  ```

- action:
  1) condition is tested
  2) if the condition is true the statement is performed; if the condition is false, continue after the loop
  3) after completing the loop’s body, go back to number 1

- **iteration** = one execution of the subsidiary statement
int var = 1;
while (var < 5) {
    printf("%d\n", var);
    var = var + 1;
}


Infinite Loops

- The loop will **never** terminate on its own.
- In UNIX, to stop a program with an infinite loop
  - Control-c
  - there can be a delay

```c
int var = 1;
while (var < 5) {
    printf("%d\n", var);
}
```
The do/while loop

- Format:
  ```
  do {
    statements
  } while (condition);
  ```
- the curly braces are not required, but good style otherwise the line with "while" can be easily confused with the beginning of a while loop.

- Action:
  1) execute the body
  2) test the condition
  3) if the condition is true, go back to #1; if the condition is false, continue with the line after the loop
sum = 0;
do {
    j--;
    sum+=j;
} while (j > 0);

---

sum= 0;
while (j > 0){
    j--;
    sum += j;
}

Same or Different?
Trace both with different starting values for j
Types of repetition:

- counter-controlled repetition
- indefinite repetition
- examples

end of exam #1 materials