

CSCA08 FALL 2017

WEEK 5 - ITERATION

Brian Harrington & Marzieh Ahmadzadeh

University of Toronto Scarborough

October 2 - 6, 2017

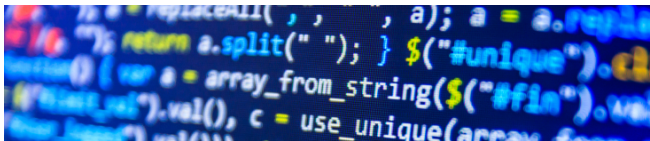


UNIVERSITY OF
TORONTO
SCARBOROUGH

ADMIN STUFF

- Term Test 1
- ex2
- A0
- Reading Week

ONTRACK 2017



CMS OnTrack Workshop 2017

"Get On Track, Stay On Track"

Learn the skills you'll need to be on track for a 4.0!

11:00am – 4:00pm

Tuesday October 10

Dept. of Computer and Mathematical Sciences



UNIVERSITY OF
TORONTO
SCARBOROUGH

<http://uoft.me/OnTrack>

WHILE LOOPS

- Simplest type of loop
- Can do anything the other types can do
 - But technically my forehead can do anything a hammer can do... that doesn't mean it's always the best tool for the job
- Form:

```
while(condition):  
    block
```
- keep executing `block` as long as `condition` evaluates to `True`
 - `condition` is only checked before each time `block` is executed, not during the execution
 - beware the **infinite loop!**

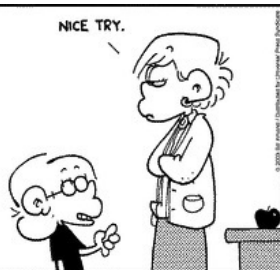
INDEXED FOR LOOPS

- Perform a series of actions a counted number of times
- Can use the number as an index into elements of a list
- Form:

```
for counter in range(start, end, skip):  
    block
```
- Execute `block` once for each number in the range, with `counter` set to that number

BREAK

```
def punishment(message):  
    for i in range(0,500):  
        print(message)  
  
punishment("I will not throw paper airplanes in class")
```



© 2008 by Andrew J. Chumbley for "Creative Commons License"

ELEMENTAL FOR LOOPS

- Perform a series of actions once for each element in a list
- Great for quickly/easily getting through an entire list of input
- Problem: We don't know where we are in the list at any given time
- Form:

```
for element in list:  
    block
```
- `block` is executed once for each value of `element`