

Advanced Functions

Doug Blank
CS110 Fall 2009
Bryn Mawr College

Functions as Arguments

```
from myro import *
```

```
def hello():  
    speak("Hello")
```

```
def goodbye():  
    speak("Goodbye")
```

```
def dialog(f1, f2):  
    f1()  
    f2()
```

```
>>> dialog(hello, goodbye)
```

Functions that stop, and don't

```
def infiniteLoop(parameter):  
    while True:  
        print "loop!"  
    return
```

```
def stops(parameter):  
    return "Hi Mom! Send money!"
```

Could you write a function that determines if a function will stop or not?

```
def halts?(function, parameter):  
    if function(parameter) halts:  
        return True  
    else:  
        return False
```

Assume that you could!

```
>>> halts?(infinteLoop, 38)
```

```
False
```

```
>>> halts?(stops, 56)
```

```
True
```

Assume that you could!

```
def G(function, parameter):  
    if halts?(function, parameter):  
        while True:  
            print "looping because it stops!"  
    else:  
        return "Returning because it loops!"
```

Assume that you could!

```
def G(function, parameter):  
    if halts?(function, parameter):  
        while True:  
            print "looping because it stops!"  
    else:  
        return "Returning because it loops!"
```

```
>>> G(G, G)
```

What will happen?

A contradiction!

Therefore, our assumption is invalid:

The function **halts?** cannot be written.

Turing's Halting Problem

