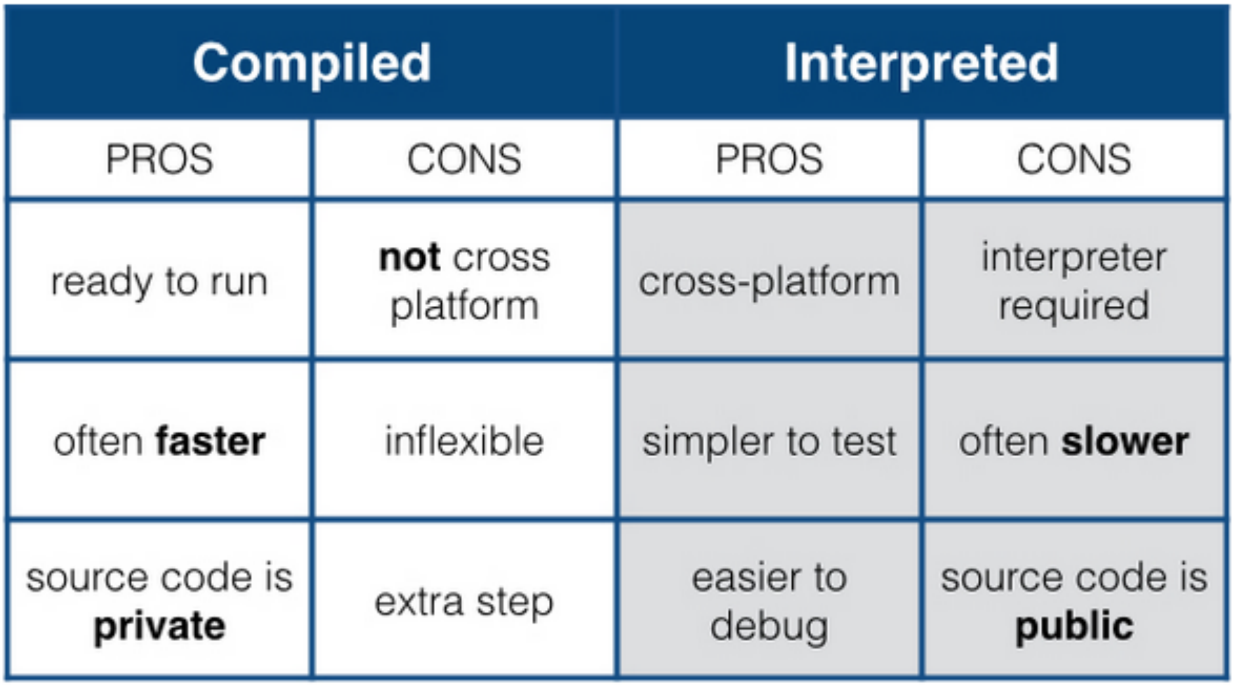
Guiding questions:

Why is Python easier to learn than other languages?

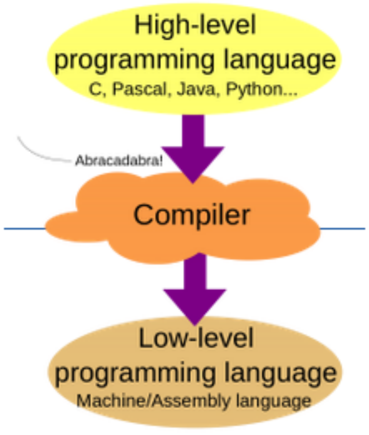
Python is a fast, flexible, beginner-friendly programming language. It's gradual learning curve and readability make it an excellent choice for launching your adventures in coding. Python is also amazingly powerful. NASA, Google and Disney, to name a few, use it for everything from web applications to robots. Take a byte of Python and quickly learn to think like a programmer with our free videos and tutorials. – From thehelloworldprogram.com/python

What is the difference between Interpreted and Compiled languages?



Compiled		Interpreted	
PROS	CONS	PROS	CONS
ready to run	**not** cross platform	cross-platform	interpreter required
often **faster**	inflexible	simpler to test	often **slower**
source code is **private**	extra step	easier to debug	source code is **public**

SOURCE CODE GETS CONVERTED BEFORE IT CAN RUN.



Here's a flow chart showing how high-level languages communicate with the computer.

There are two main ways to convert: compiled vs interpreted. Luckily, it's not a decision we have to worry about. But it is worth knowing difference; however, don't get too bogged down in the details.

Compiled languages is when a person writes the code, compiler separates the file and the end result is an executable file. Basically, owner keeps the source code.

Interpreted languages are different because the code is not compiled first hand. Instead, a copy is given to another machine and that machine interprets it. An example is JavaScript (which is used everywhere online).

Where are the Python programs in Pippy stored on the OLPC directory structure?

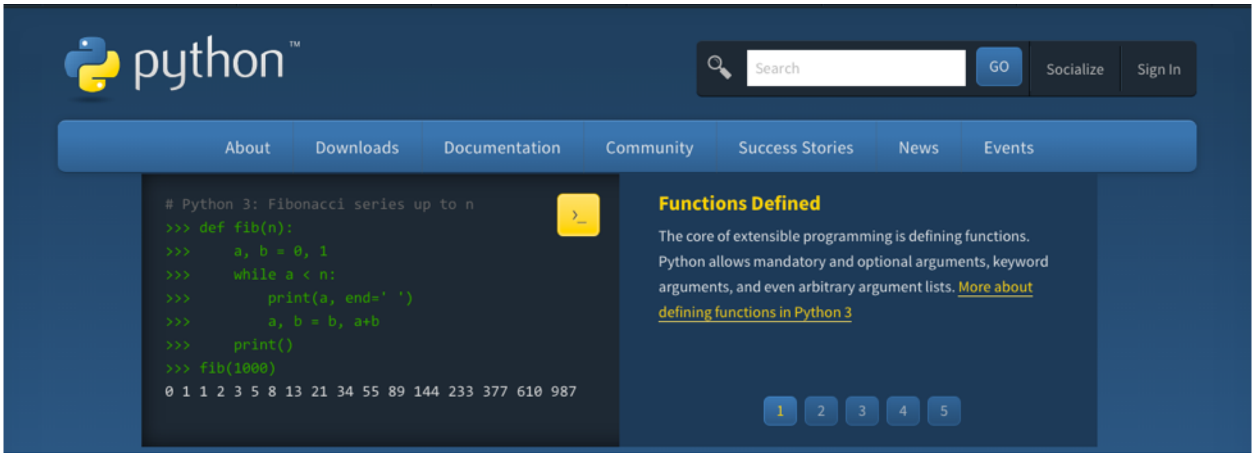
/home/olpc/Activities/Pippy.activity/data

Vocabulary to define:

Python: A versatile high-level programming language which uses an interpreter. Used in many varieties.
Pippy : A interface to help kids program with Python; commonly called an IDE
Interpreted: An interpreted program is basically a language which requires an interpreter in-OS; instead of exporting compiled code, an interpreter converts the source code on machine.
Compiled : Compiled programs are a language, which, unlike Interpreted programs, exports a .exe/.app/.jar etc. which does not require an interpreter to run. Instead, the program requires the use of a compiler to compile the code and create the extension'd file.

Tasks:

1. Log into the UCC network using your MacBook and the UCC1 access point (Done!)
2. Type "what is the python programming language" into Google
3. Surf to the main page of python.org and take a screenshot

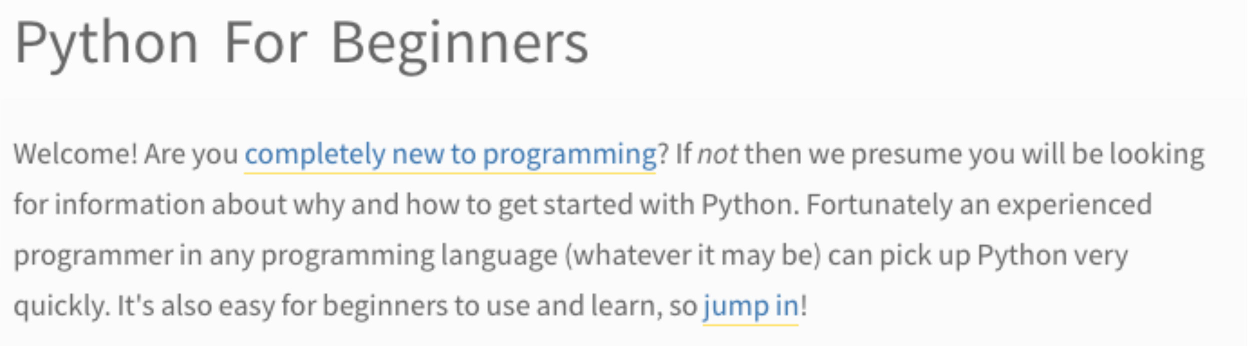


```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

Functions Defined

The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. More about defining functions in Python 3

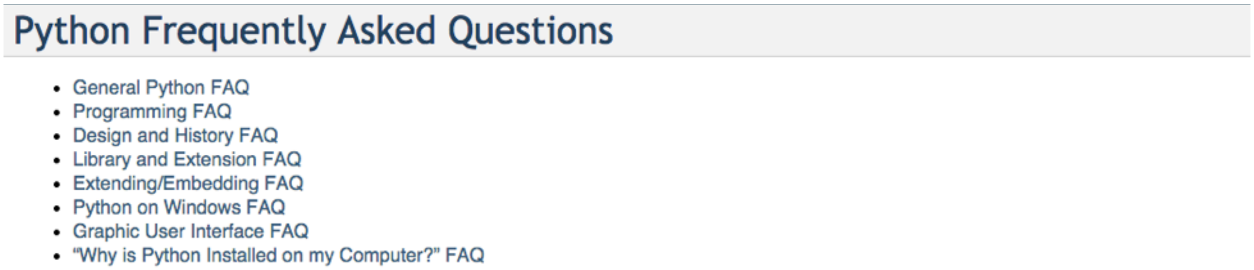
4. Surf to the Python Info Wiki Beginner's Guide (and take a screenshot)



Python For Beginners

Welcome! Are you completely new to programming? If *not* then we presume you will be looking for information about why and how to get started with Python. Fortunately an experienced programmer in any programming language (whatever it may be) can pick up Python very quickly. It's also easy for beginners to use and learn, so jump in!

5. Surf to the Python FAQ site (and take a screenshot)

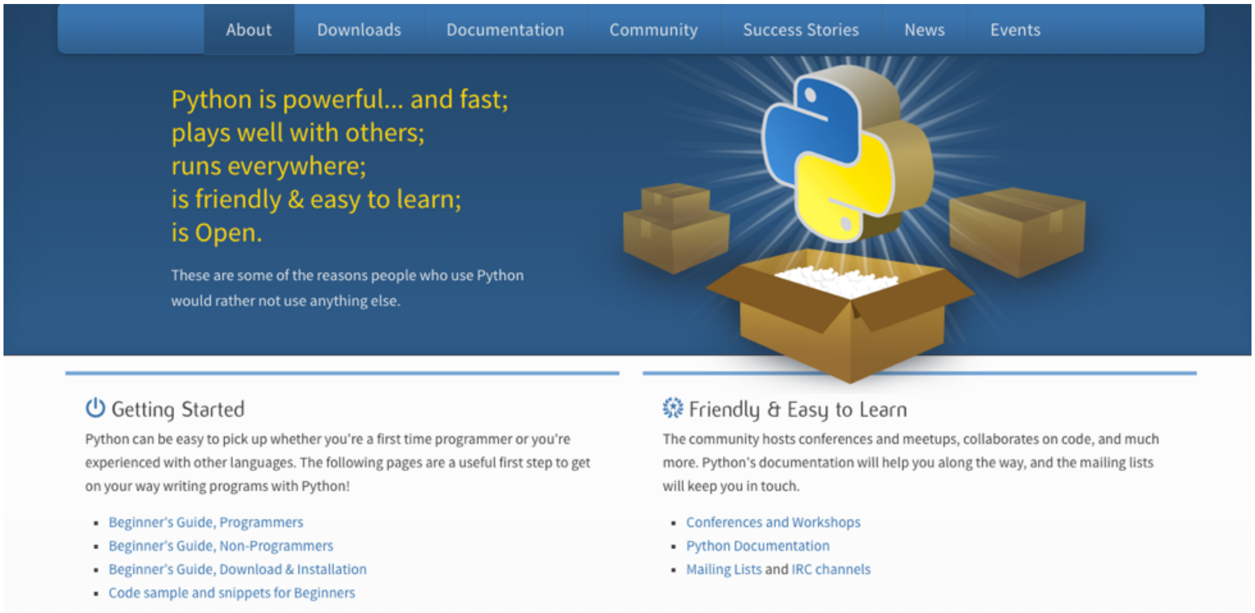


Python Frequently Asked Questions

- General Python FAQ
- Programming FAQ
- Design and History FAQ
- Library and Extension FAQ
- Extending/Embedding FAQ
- Python on Windows FAQ
- Graphic User Interface FAQ
- "Why is Python Installed on my Computer?" FAQ

---

6. Surf to the About Python page (and take a screenshot)



Python is powerful... and fast;
plays well with others;
runs everywhere;
is friendly & easy to learn;
is Open.

These are some of the reasons people who use Python would rather not use anything else.

Getting Started

Python can be easy to pick up whether you're a first time programmer or you're experienced with other languages. The following pages are a useful first step to get on your way writing programs with Python!

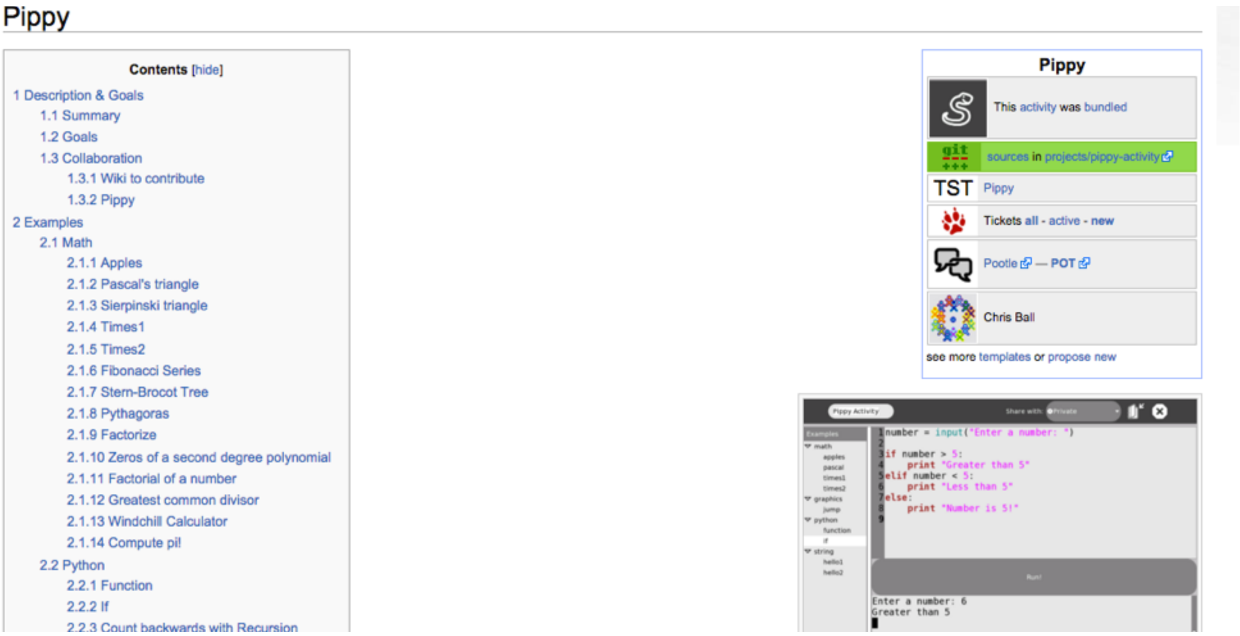
- Beginner's Guide, Programmers
- Beginner's Guide, Non-Programmers
- Beginner's Guide, Download & Installation
- Code sample and snippets for Beginners

Friendly & Easy to Learn

The community hosts conferences and meetups, collaborates on code, and much more. Python's documentation will help you along the way, and the mailing lists will keep you in touch.

- Conferences and Workshops
- Python Documentation
- Mailing Lists and IRC channels

7. Type Pippy XO into Google and surf to the Pippy wiki page (and take a screenshot)



Pippy

Contents [hide]

Pippy

This activity was bundled

sources in projects/pippy-activity

TST Pippy

Tickets all - active - new

Pootle — POT

Chris Ball

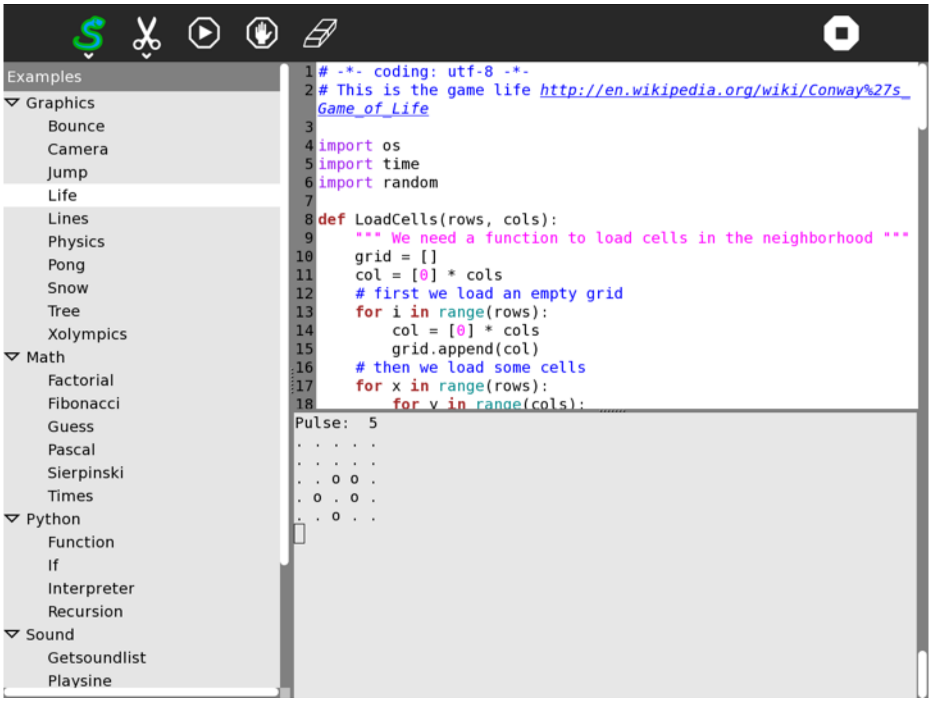
see more templates or propose new

```
number = input("Enter a number: ")

if number > 5:
    print "Greater than 5"
elif number < 5:
    print "Less than 5"
else:
    print "Number is 5!"

```

```
Enter a number: 6
Greater than 5
```

8. Receive an OLPX XO 1.5 and turn it on
9. Find and Launch the Pippy App on your OLPC XO

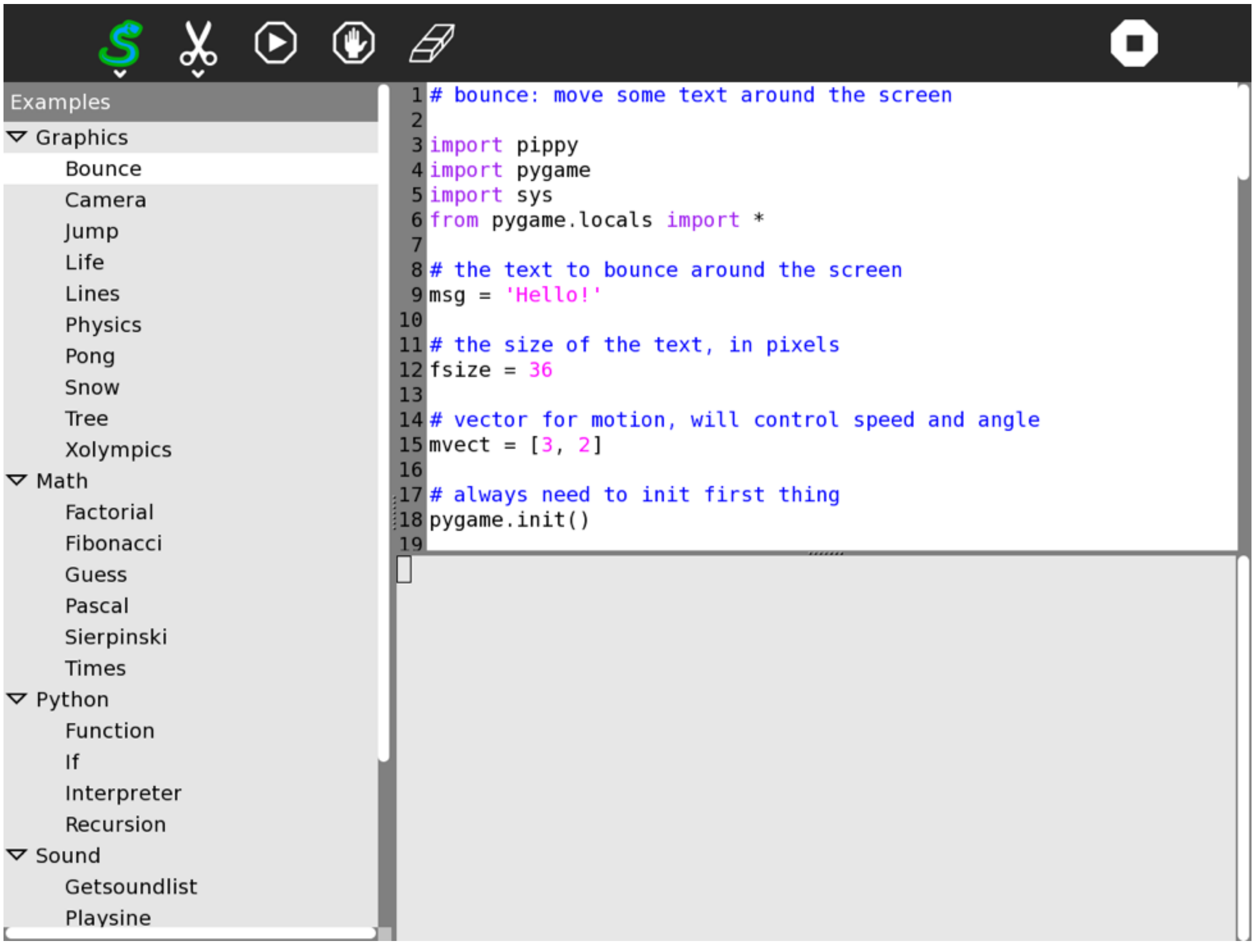
10. Interact with the Pippy Development Environment to figure out how it works (taking screenshots as you go)



```
# -*- coding: utf-8 -*-
# This is the game life http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life

import os
import time
import random

def LoadCells(rows, cols):
    """ We need a function to load cells in the neighborhood """
    grid = []
    col = [0] * cols
    # first we load an empty grid
    for i in range(rows):
        col = [0] * cols
        grid.append(col)
    # then we load some cells
    for x in range(rows):
        for y in range(cols):
```

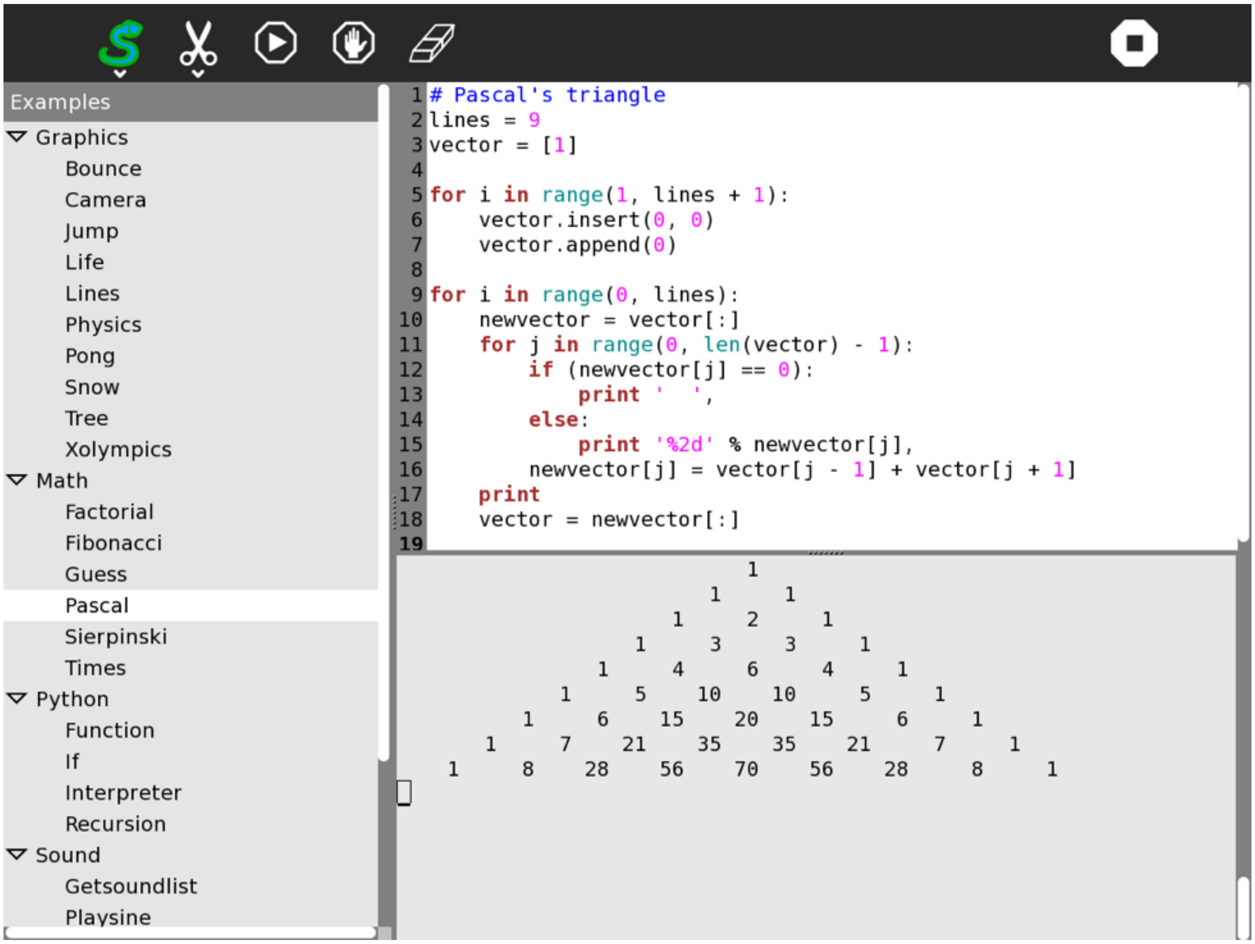
```
Pulse:  5
. . . . .
. . . . .
. . o o .
. o . o .
. . o . .
```

11. Try out some of the programs (taking screenshots as you go).



```
# bounce: move some text around the screen

import pippy
import pygame
import sys
from pygame.locals import *

# the text to bounce around the screen
msg = 'Hello!'

# the size of the text, in pixels
fsize = 36

# vector for motion, will control speed and angle
mvect = [3, 2]

# always need to init first thing
pygame.init()

```

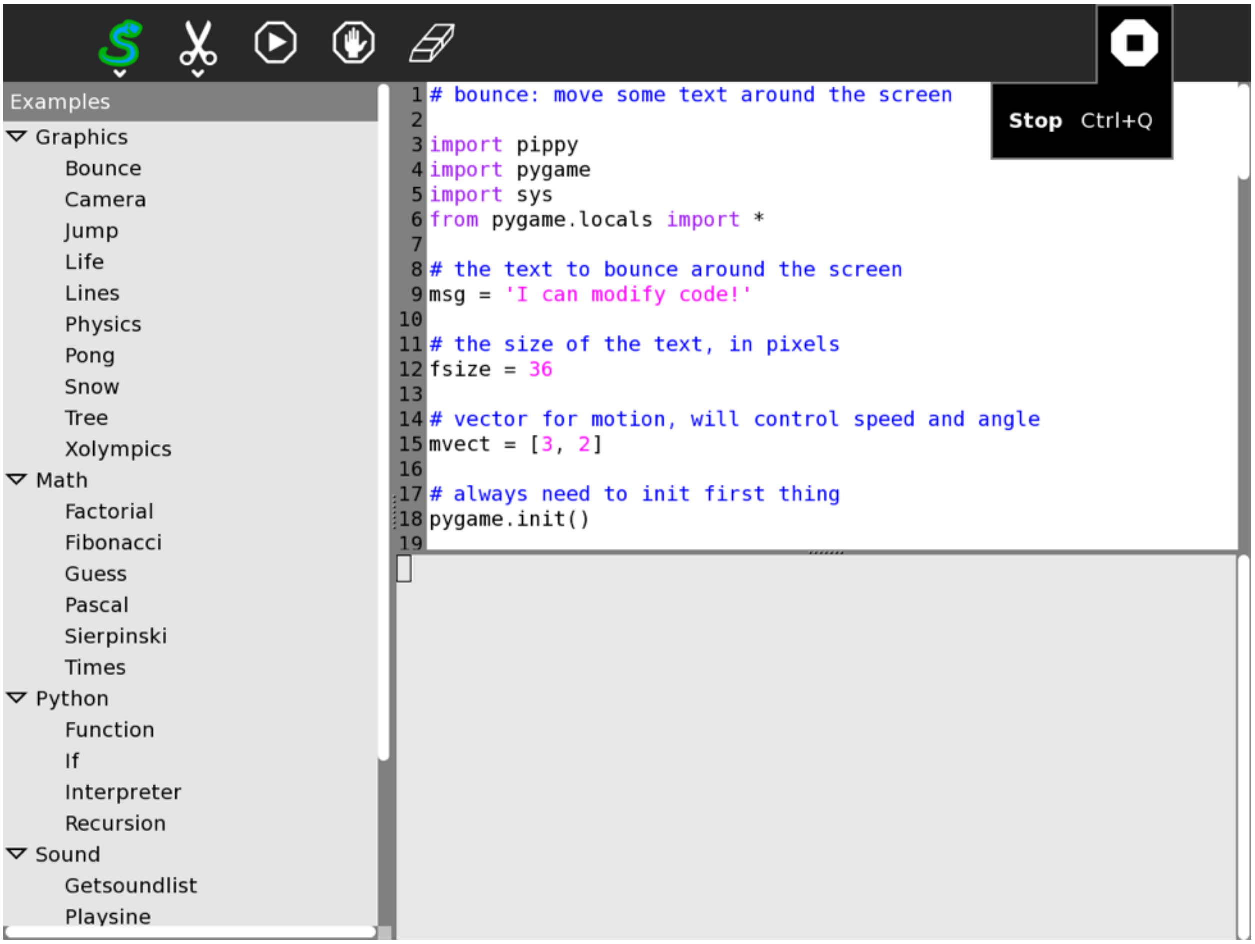
12. Are there are different categories of sample programs in Pippy? Try the math samples.



```
# Pascal's triangle
lines = 9
vector = [1]

for i in range(1, lines + 1):
    vector.insert(0, 0)
    vector.append(0)

for i in range(0, lines):
    newvector = vector[:]
    for j in range(0, len(vector) - 1):
        if (newvector[j] == 0):
            print '  ',
        else:
            print '%2d' % newvector[j],
        newvector[j] = vector[j - 1] + vector[j + 1]
    print
    vector = newvector[:]
```

```
                         1
                      1     1
                   1     2     1
                1     3     3     1
             1     4     6     4     1
          1     5    10    10     5     1
       1     6    15    20    15     6     1
    1     7    21    35    35    21     7     1
 1     8    28    56    70    56    28     8     1
```

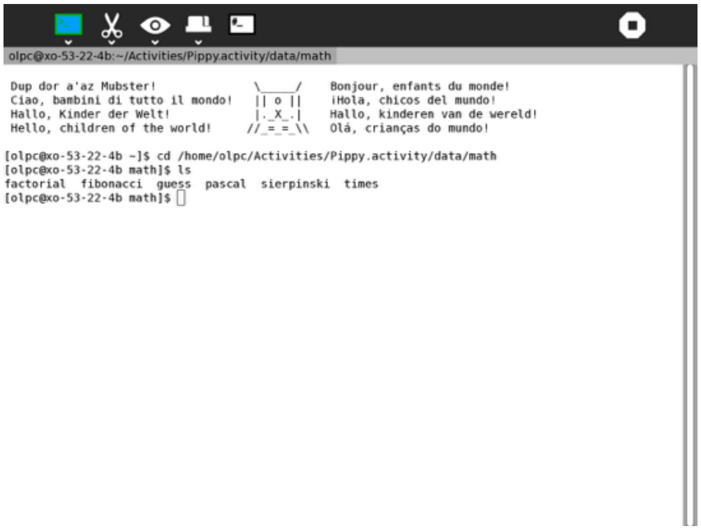
13. Can you use Pippy to save a Python program that you modified? Take screenshots and explain.



```python
# bounce: move some text around the screen

import pippy
import pygame
import sys
from pygame.locals import *

# the text to bounce around the screen
msg = 'I can modify code!'

# the size of the text, in pixels
fsize = 36

# vector for motion, will control speed and angle
mvect = [3, 2]

# always need to init first thing
pygame.init()

```

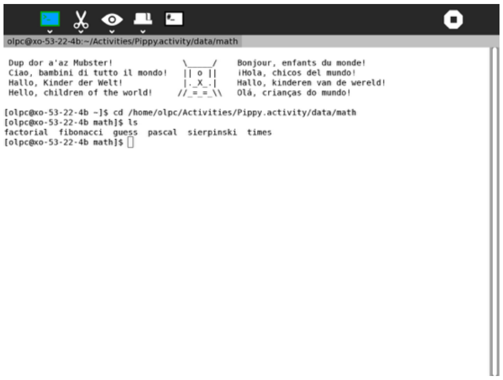
14. Open a terminal and cd to /home/olpc/Activities/Pippy.activity/data/math
Use the Linux ls command to list the files in this directory

15.

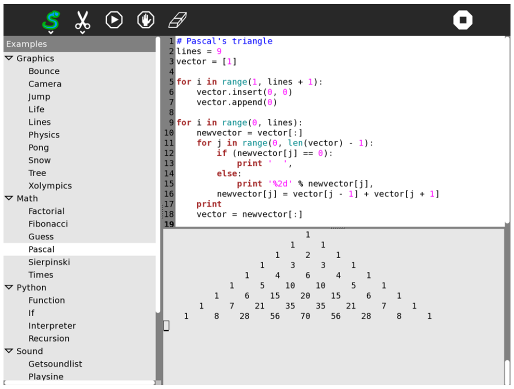


```
 Dup dor a'az Mubster!            \_____/    Bonjour, enfants du monde!
 Ciao, bambini di tutto il mondo!  || o ||    ¡Hola, chicos del mundo!
 Hallo, Kinder der Welt!           |._X_.|    Hallo, kinderen van de wereld!
 Hello, children of the world!    //_=_=_\\   Olá, crianças do mundo!

[olpc@xo-53-22-4b ~]$ cd /home/olpc/Activities/Pippy.activity/data/math
[olpc@xo-53-22-4b math]$ ls
factorial  fibonacci  guess  pascal  sierpinski  times
[olpc@xo-53-22-4b math]$ 
```

16. Take a screen shot and compare this list to the math sample List in Pippy



```
olpc@xo-53-22-4b:~/Activities/Pippy.activity/data/math

Dup dor a'az Mubster!              \_____/    Bonjour, enfants du monde!
Ciao, bambini di tutto il mondo!   || o ||    ¡Hola, chicos del mundo!
Hallo, Kinder der Welt!            |._X_.|    Hallo, kinderen van de wereld!
Hello, children of the world!     //_=_=_\\   Olá, crianças do mundo!

[olpc@xo-53-22-4b ~]$ cd /home/olpc/Activities/Pippy.activity/data/math
[olpc@xo-53-22-4b math]$ ls
factorial  fibonacci  guess  pascal  sierpinski  times
[olpc@xo-53-22-4b math]$ 
```

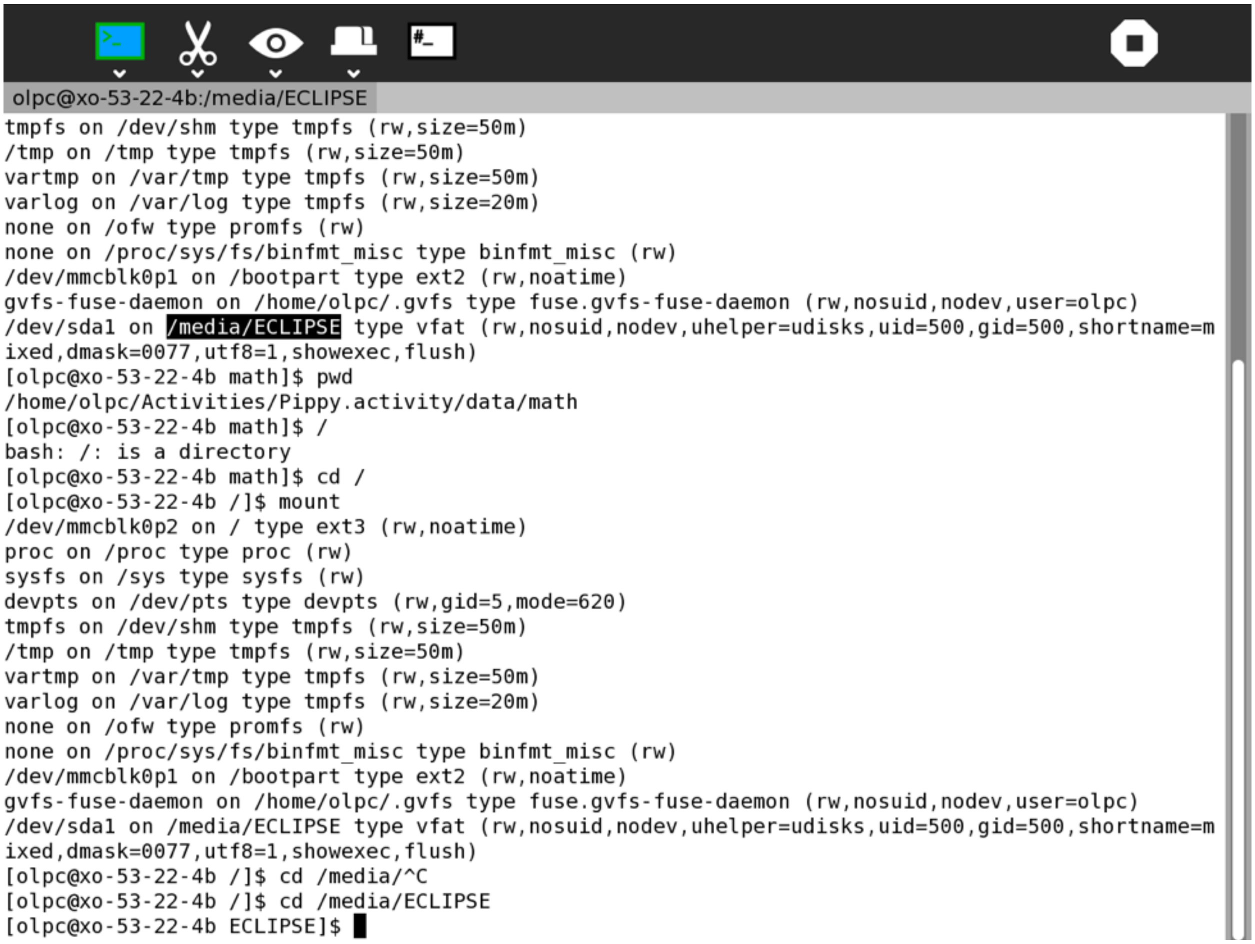


```
# Pascal's triangle
lines = 9
vector = [1]

for i in range(1, lines + 1):
    vector.insert(0, 0)
    vector.append(0)

for i in range(0, lines):
    newvector = vector[:]
    for j in range(0, len(vector) - 1):
        if (newvector[j] == 0):
            print '  ',
        else:
            print '%2d' % newvector[j],
        newvector[j] = vector[j - 1] + vector[j + 1]
    print
    vector = newvector[:]
```

```
                            1
                         1     1
                      1     2     1
                   1     3     3     1
                1     4     6     4     1
             1     5    10    10     5     1
          1     6    15    20    15     6     1
       1     7    21    35    35    21     7     1
    1     8    28    56    70    56    28     8     1
```

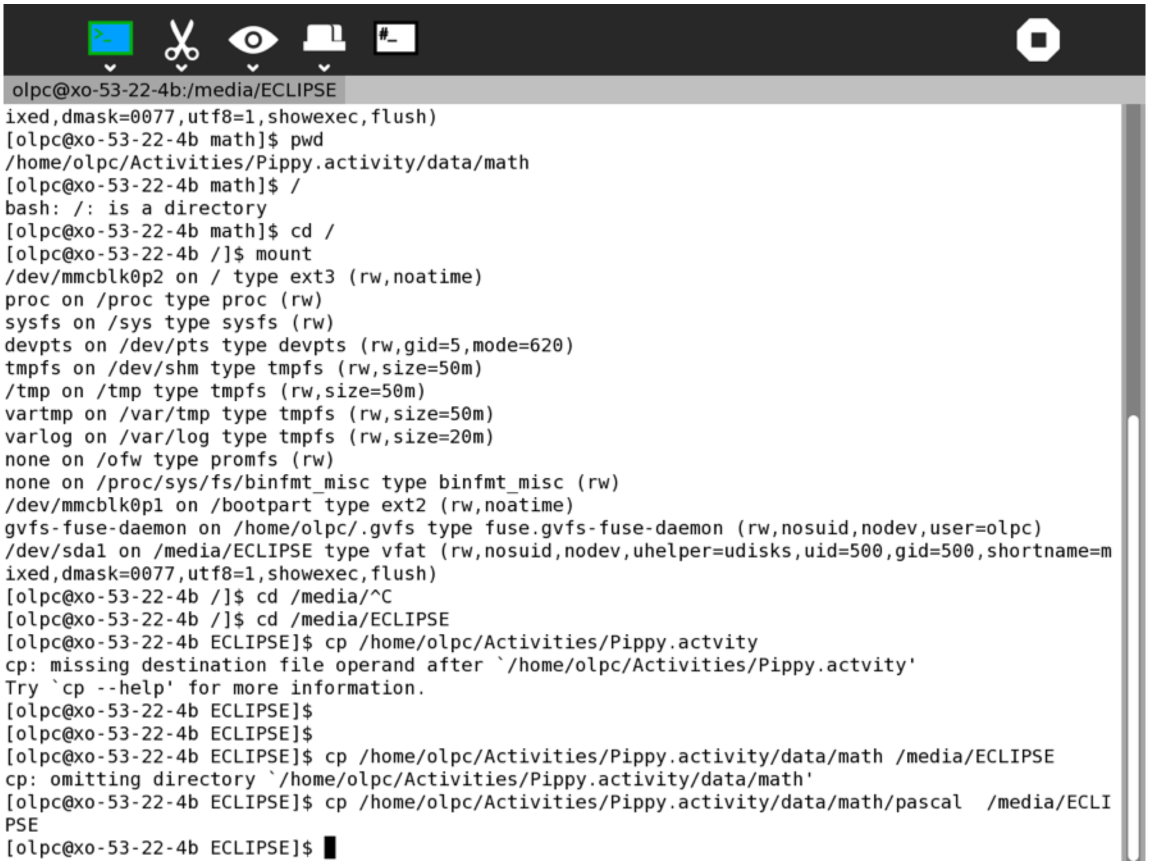
17. Receive a USB stick and insert it into your OLPC 1.5

18. Use the Linux mount command to find the path name of the USB stick. Take a screenshot.



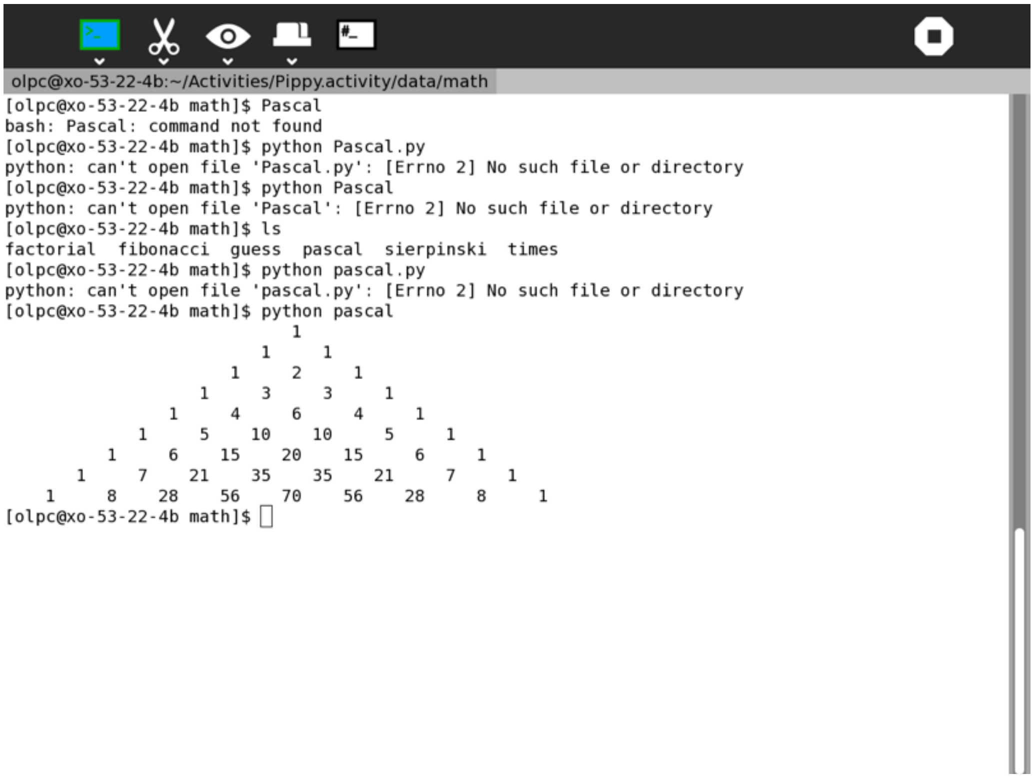
```
olpc@xo-53-22-4b:/media/ECLIPSE
tmpfs on /dev/shm type tmpfs (rw,size=50m)
/tmp on /tmp type tmpfs (rw,size=50m)
vartmp on /var/tmp type tmpfs (rw,size=50m)
varlog on /var/log type tmpfs (rw,size=20m)
none on /ofw type promfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
/dev/mmcblk0p1 on /bootpart type ext2 (rw,noatime)
gvfs-fuse-daemon on /home/olpc/.gvfs type fuse.gvfs-fuse-daemon (rw,nosuid,nodev,user=olpc)
/dev/sda1 on /media/ECLIPSE type vfat (rw,nosuid,nodev,uhelper=udisks,uid=500,gid=500,shortname=m
ixed,dmask=0077,utf8=1,showexec,flush)
[olpc@xo-53-22-4b math]$ pwd
/home/olpc/Activities/Pippy.activity/data/math
[olpc@xo-53-22-4b math]$ /
bash: /: is a directory
[olpc@xo-53-22-4b math]$ cd /
[olpc@xo-53-22-4b /]$ mount
/dev/mmcblk0p2 on / type ext3 (rw,noatime)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw,size=50m)
/tmp on /tmp type tmpfs (rw,size=50m)
vartmp on /var/tmp type tmpfs (rw,size=50m)
varlog on /var/log type tmpfs (rw,size=20m)
none on /ofw type promfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
/dev/mmcblk0p1 on /bootpart type ext2 (rw,noatime)
gvfs-fuse-daemon on /home/olpc/.gvfs type fuse.gvfs-fuse-daemon (rw,nosuid,nodev,user=olpc)
/dev/sda1 on /media/ECLIPSE type vfat (rw,nosuid,nodev,uhelper=udisks,uid=500,gid=500,shortname=m
ixed,dmask=0077,utf8=1,showexec,flush)
[olpc@xo-53-22-4b /]$ cd /media/^C
[olpc@xo-53-22-4b /]$ cd /media/ECLIPSE
[olpc@xo-53-22-4b ECLIPSE]$ █
```

19. Challenge 1: Use a Linux command to copy the Pascal Pippy program to your USB and then your Mac. Take a screenshot.



```
ixed,dmask=0077,utf8=1,showexec,flush)
[olpc@xo-53-22-4b math]$ pwd
/home/olpc/Activities/Pippy.activity/data/math
[olpc@xo-53-22-4b math]$ /
bash: /: is a directory
[olpc@xo-53-22-4b math]$ cd /
[olpc@xo-53-22-4b /]$ mount
/dev/mmcblk0p2 on / type ext3 (rw,noatime)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw,size=50m)
/tmp on /tmp type tmpfs (rw,size=50m)
vartmp on /var/tmp type tmpfs (rw,size=50m)
varlog on /var/log type tmpfs (rw,size=20m)
none on /ofw type promfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
/dev/mmcblk0p1 on /bootpart type ext2 (rw,noatime)
gvfs-fuse-daemon on /home/olpc/.gvfs type fuse.gvfs-fuse-daemon (rw,nosuid,nodev,user=olpc)
/dev/sda1 on /media/ECLIPSE type vfat (rw,nosuid,nodev,uhelper=udisks,uid=500,gid=500,shortname=m
ixed,dmask=0077,utf8=1,showexec,flush)
[olpc@xo-53-22-4b /]$ cd /media/^C
[olpc@xo-53-22-4b /]$ cd /media/ECLIPSE
[olpc@xo-53-22-4b ECLIPSE]$ cp /home/olpc/Activities/Pippy.actvity
cp: missing destination file operand after `/home/olpc/Activities/Pippy.actvity'
Try `cp --help' for more information.
[olpc@xo-53-22-4b ECLIPSE]$
[olpc@xo-53-22-4b ECLIPSE]$
[olpc@xo-53-22-4b ECLIPSE]$ cp /home/olpc/Activities/Pippy.activity/data/math /media/ECLIPSE
cp: omitting directory `/home/olpc/Activities/Pippy.activity/data/math'
[olpc@xo-53-22-4b ECLIPSE]$ cp /home/olpc/Activities/Pippy.activity/data/math/pascal  /media/ECLI
PSE
[olpc@xo-53-22-4b ECLIPSE]$
```

20. Challenge 2: Run Pascal from a terminal on your Mac. Are there Take a screenshot.



```
[olpc@xo-53-22-4b math]$ Pascal
bash: Pascal: command not found
[olpc@xo-53-22-4b math]$ python Pascal.py
python: can't open file 'Pascal.py': [Errno 2] No such file or directory
[olpc@xo-53-22-4b math]$ python Pascal
python: can't open file 'Pascal': [Errno 2] No such file or directory
[olpc@xo-53-22-4b math]$ ls
factorial  fibonacci  guess  pascal  sierpinski  times
[olpc@xo-53-22-4b math]$ python pascal.py
python: can't open file 'pascal.py': [Errno 2] No such file or directory
[olpc@xo-53-22-4b math]$ python pascal
                             1
                          1     1
                       1     2     1
                    1     3     3     1
                 1     4     6     4     1
              1     5    10    10     5     1
           1     6    15    20    15     6     1
        1     7    21    35    35    21     7     1
     1     8    28    56    70    56    28     8     1
[olpc@xo-53-22-4b math]$
```

21. Challenge 3: Research and answer the GQs and vocab.
22. Create a beautiful PDF document of your scavenger hunt full of annotation and comments and upload to Haiku dropbox.