

Game tools

- second coursework is to produce a level editor for a tablet
- we will be exploring many aspects of tools
 - python
 - python pygame
 - touchgui
 - [touchmap](#)
 - debian package management
 - GNU/Linux build tools

slide 3
gaius

Touchmap

- a tablet tool for creating maps for:
 - penguin tower 2D
 - isometric penguin tower
 - doom3
- you will start with a prototype `touchmap`
 - and extend it in a variety of methods

slide 4
gaius

Touchmap

- is written in Python and uses the library `touchgui`
- you can download the `touchmap` and `touchgui` source code
 - you can change the `touchgui` code if you wish

Touchgui

- a small gui for a tablet written in Python using Pygame
- [touchgui documentation](http://floppsie.comp.glam.ac.uk/touchgui/homepage.html) (http://floppsie.comp.glam.ac.uk/touchgui/homepage.html)
- try out the [examples](http://floppsie.comp.glam.ac.uk/touchgui/examples.html) (http://floppsie.comp.glam.ac.uk/touchgui/examples.html)

Obtaining touchmap

- you can obtain the skeleton touchmap using the terminal and the command line

```
$ cd
$ mkdir -p Sandpit
$ wget http://floppsie.comp.glam.ac.uk/download/targz/touchmap-0.1.tar.gz
$ tar xzf touchmap-0.1.tar.gz
$ ls touchmap-0.1
```

- some of these commands act silently (as is the UNIX default)
 - write down a summary of each command line (hint use man)

Obtaining touchmap

- in the directory touchmap-0.1 there are a number of files
 - touchmap.py is the main program
 - there are a number of simpler versions of the same program touchmap-1.py, touchmap-2.py, touchmap-3.py and touchmap-4.py
- these simpler versions might be interesting to examine as they show how the main program was developed
 - they also serve to show how the GUI touchgui works

Building touchmap in the University laboratories

- you will need to have downloaded touchmap as above before attempting these command here
- to build touchmap, you should use the command line terminal:

```
$ cd $HOME/Sandpit
$ rm -rf build-touchmap
$ mkdir build-touchmap
$ cd build-touchmap
$ ../touchmap-0.1/configure
$ make
```

- again make a summary of these command lines in a notebook (see man)

Building touchmap in the University laboratories

- these commands build some of the map assets
 - in particular the door, wall png images which are used by touchmap
- **note that anything placed inside the directory build-touchmap is temporary**
 - examine the contents of build-touchmap (we can see the png images for walls and doors)
 - these images are built from groff scripts!

Output from the build

- ```
$../touchmap-0.1/configure
checking for a BSD-compatible install... /usr/bin/install
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether make supports nested variables... yes
...
```

## Output from the build

- ```
$ make
groff -I../touchmap-0.1 < ../touchmap-0.1/doorh.ms > t.ps
echo "" | gs -q -dBATCH -dSAFER -sDEVICE=pngmraw -r100 -sO
>>showpage, press <return> to continue<<
pnmcrop t.pnm > doorh.pnm
rm -f t.ps t.pnm
pnmscale -xsize=100 -ysize=100 doorh.pnm > t.pnm
pnmtopng t.pnm > doorh.png
pnmtopng: 1 colors found
rm -f t.pnm
...
```

Running touchmap

- ```
$ cd $HOME/Sandpit/build-touchmap
$./localrun.sh touchmap.py
```
- notice the capital S for Sandpit
- notice the ./ before localrun.sh
  - this is required as it means use localrun.sh in this directory

## Controlling touchmap

- the quit button at the top right, terminate the program
- the return button at the top left, displays the current map (to stdout)
- the expand and shrink buttons zoom in and out of the map

## Controlling touchmap

- you can create a map by clicking in the middle of the screen
  - a tap will create a wall, a further single tap converts a wall into a door
  - another tap will convert a door into a space
- a double tap defines the start of a wall
  - the next wall square created along a straight line will create a long wall (rather than an individual block)

## Tutorial

- download and build `touchmap` using the commands from the lecture
- run `touchmap`
- explore the files in the `touchmap` package directory and make notes on the function of each file

## Bug in touchgui

- you will need to perform this command in the terminal before you run `touchgui` for the first time
- ```
mkdir -p $HOME/.cache/touchgui
```
- this command creates the directories:
 - `.cache/touchgui` in your `$HOME`
 - this directory is used as an image cache by `touchgui`

