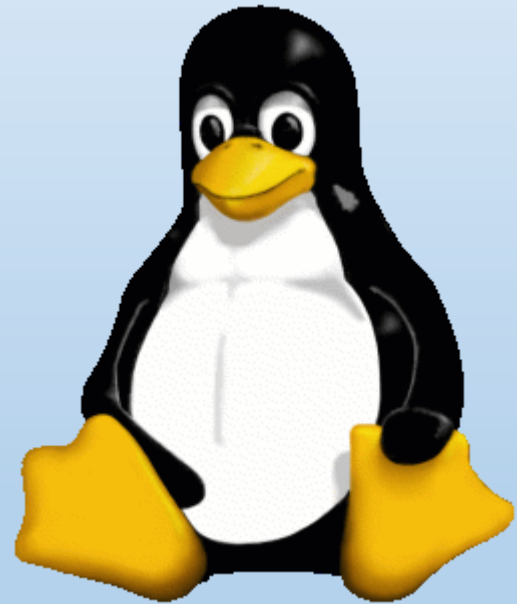


Linux Filesystem



What is this?

A series of mini-games to play with the file system navigation



File system trees

Path maniac:

Follow the commands and draw the results

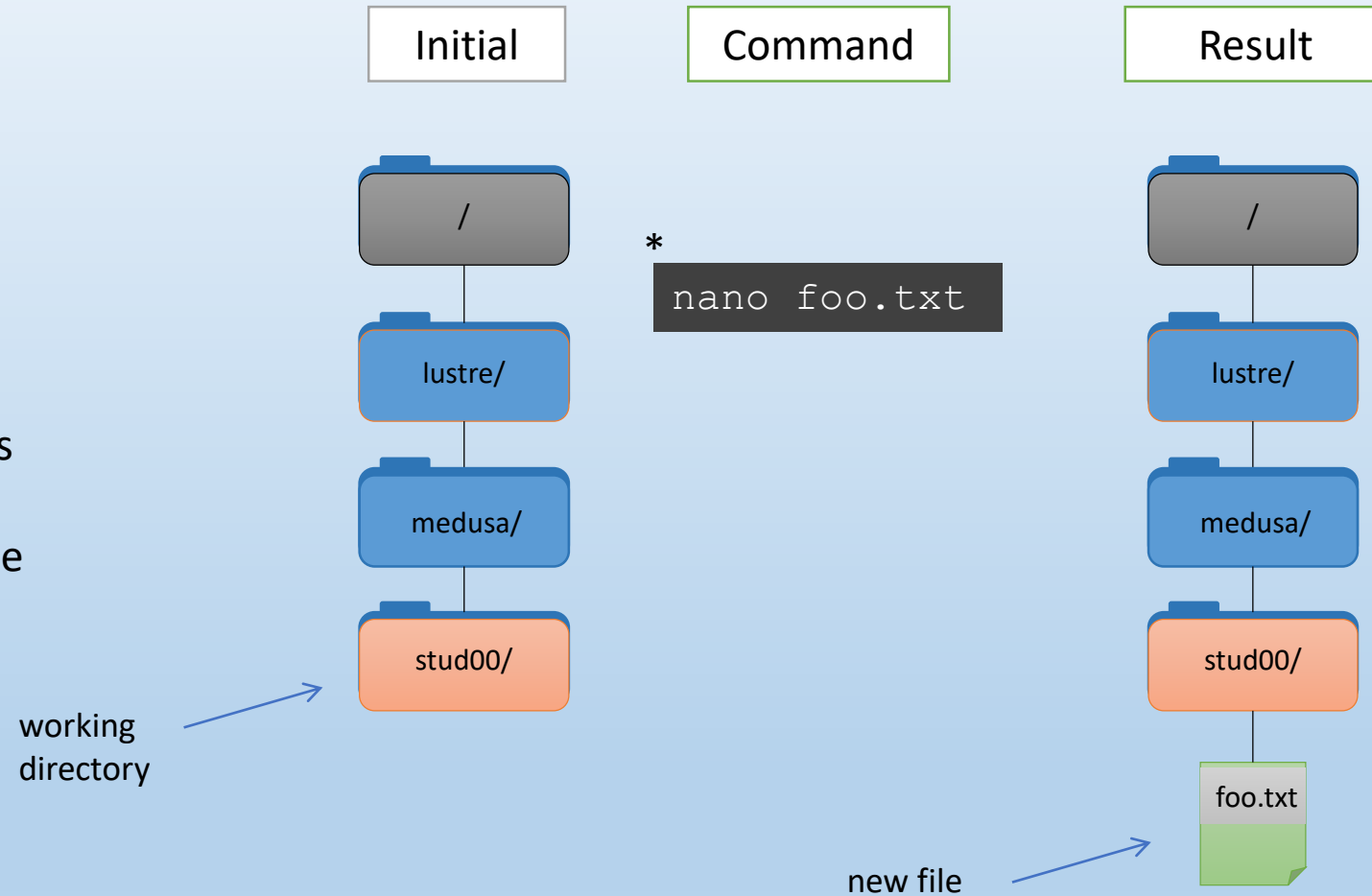
How does it work?

You will be given:

1. A path tree
2. A series of commands

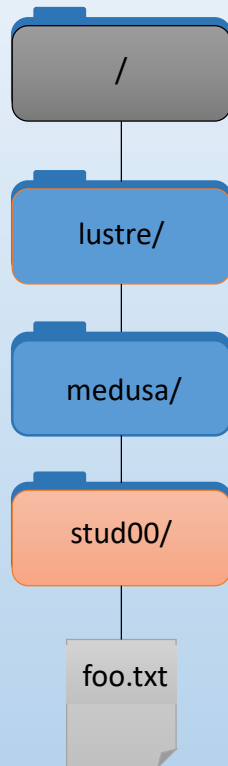
Your mission:

- Draw the resulting tree



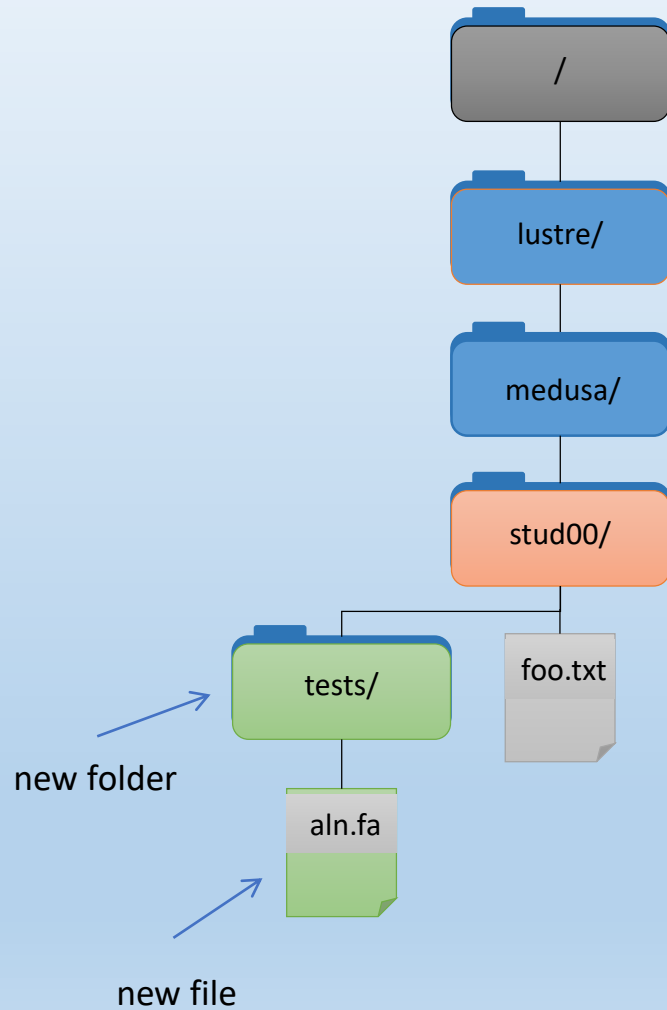
* Remember to type and save some text into nano to create the file

Path maniac:
Follow the commands and draw the results



```
mkdir tests  
nano tests/aln.fa
```

Path maniac:
Follow the commands and draw the results

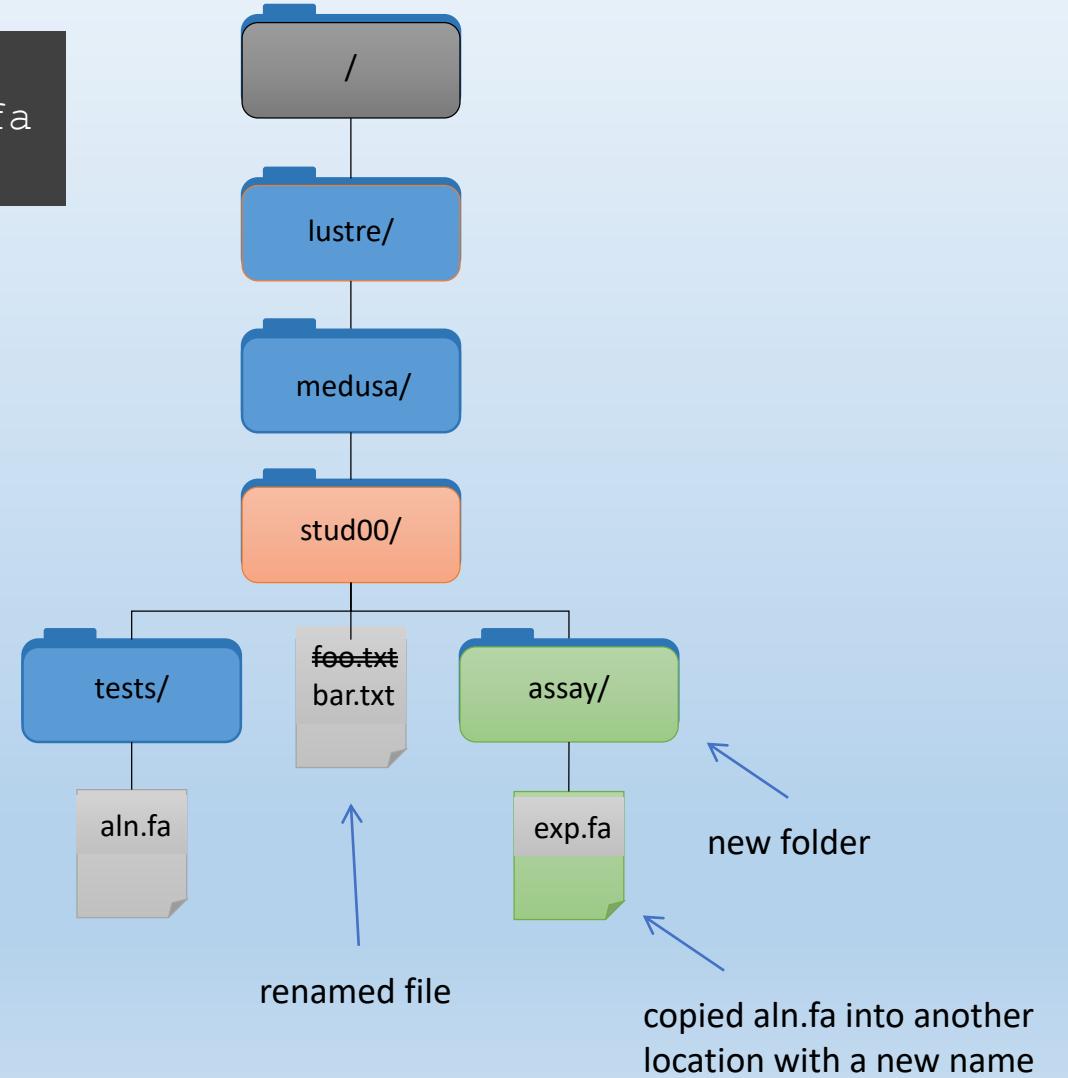
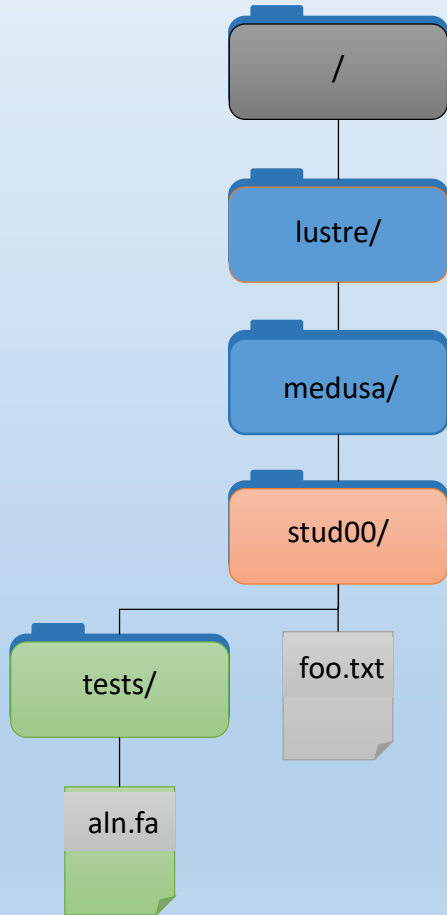


```
mkdir assay  
cp tests/aln.fa assay/exp.fa  
mv foo.txt bar.txt
```

Path maniac:

Follow the commands and draw the results

```
mkdir assay  
cp tests/aln.fa assay/exp.fa  
mv foo.txt bar.txt
```



Absolute path names

Absolute advantage:

Write the Shell commands that generate the final tree from the initial tree using only absolute paths

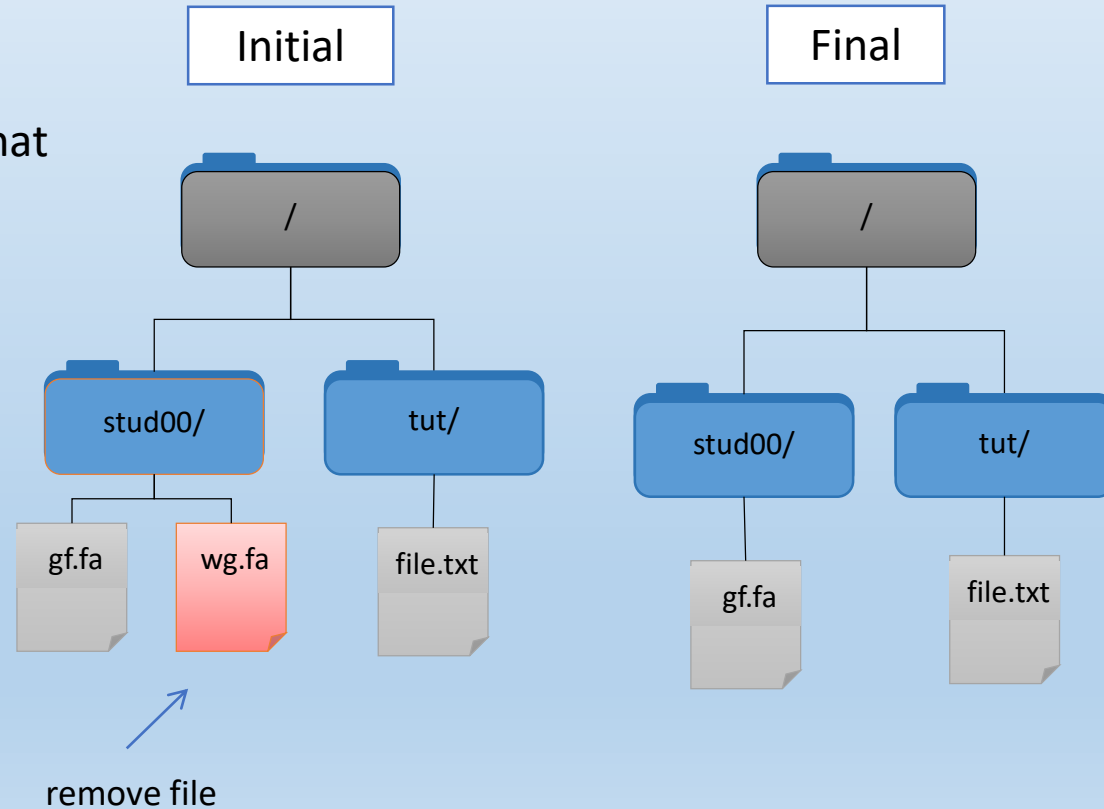
How does it work?

You will be given:

1. An initial path tree
2. A final path tree

Your mission:

- Write the commands that make the change using absolute paths

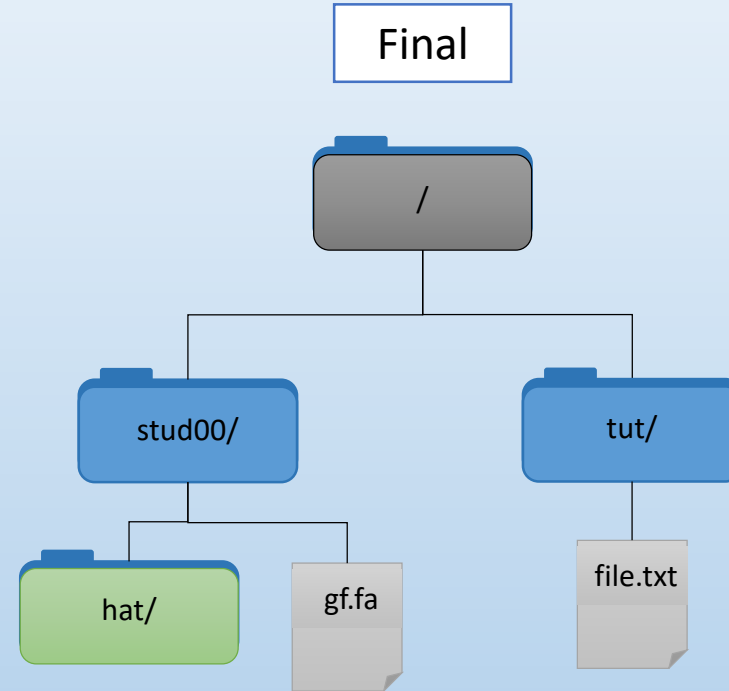
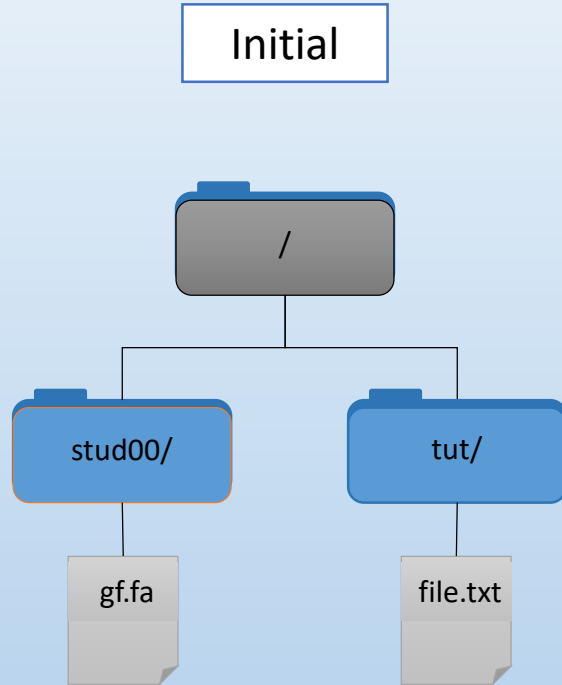


Command

```
rm /stud00/wg.fa
```

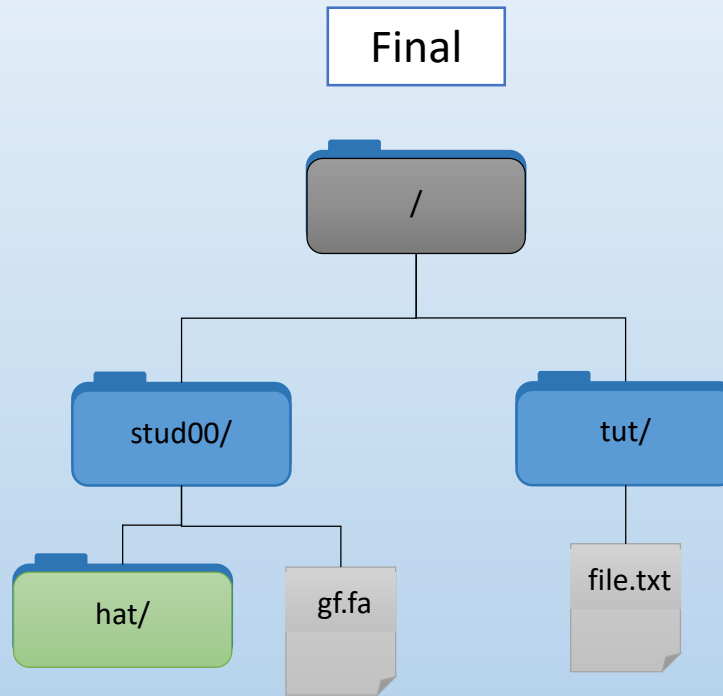
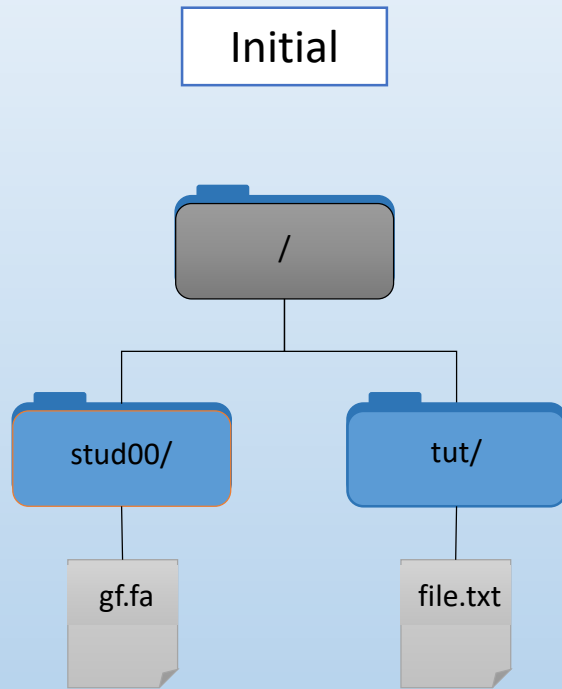
Absolute advantage challenge:

Write the Shell commands that generate the final tree from the initial tree using only absolute paths



Absolute advantage challenge:

Write the Shell commands that generate the final tree from the initial tree using only absolute paths

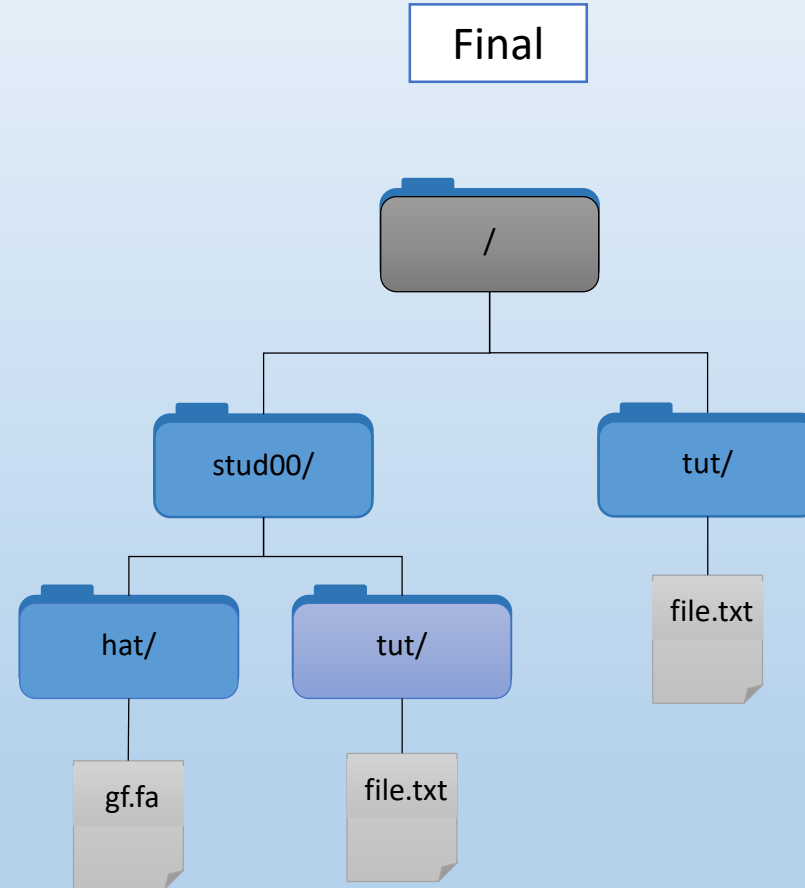
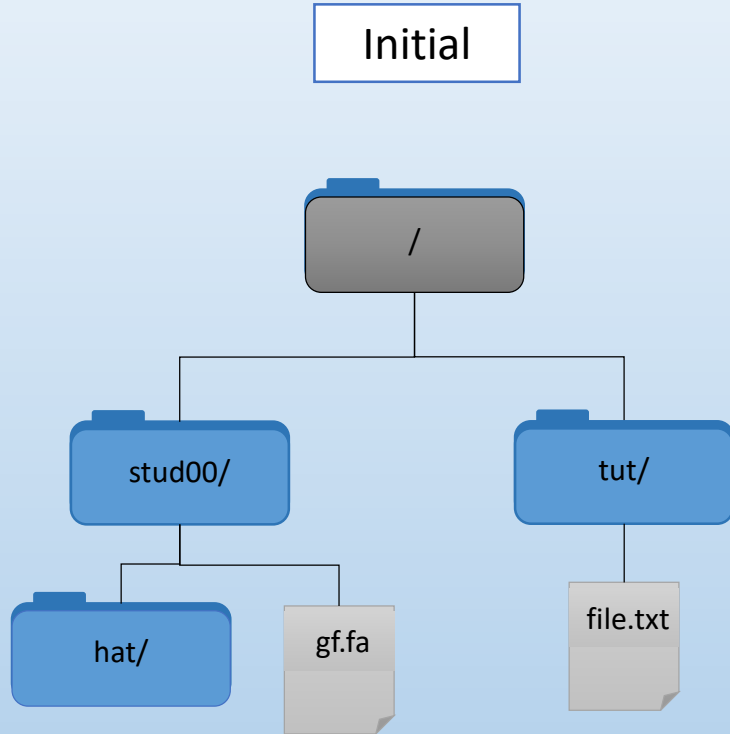


Command

```
mkdir /stud00/hat
```

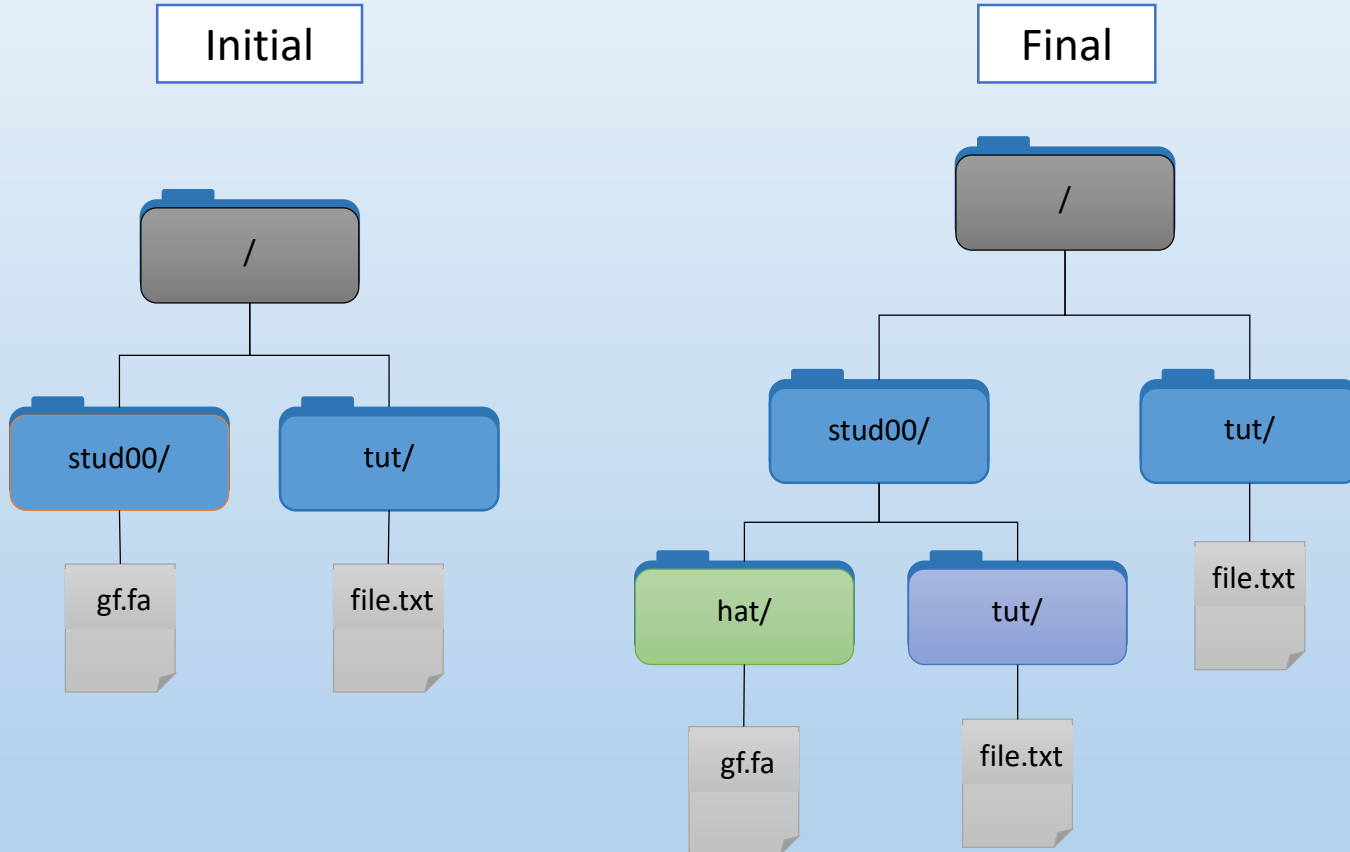
Absolute advantage challenge:

Write the Shell commands that generate the final tree from the initial tree using only absolute paths



Absolute advantage challenge :

Write the Shell commands that generate the final tree from the initial tree using only absolute paths



Commands

```
mv /stud00/gf.fa /stud00/hat  
cp -r /rnaseq_ws/tut/ /stud00/
```

Another option is

```
mkdir tut  
cp /tut/file.txt /stud00/tut  
mv /stud00/gf.fa /stud00/hat
```

Relative path names

Special relativity:

Write the Shell commands that generate the final tree from the initial tree using only relative paths

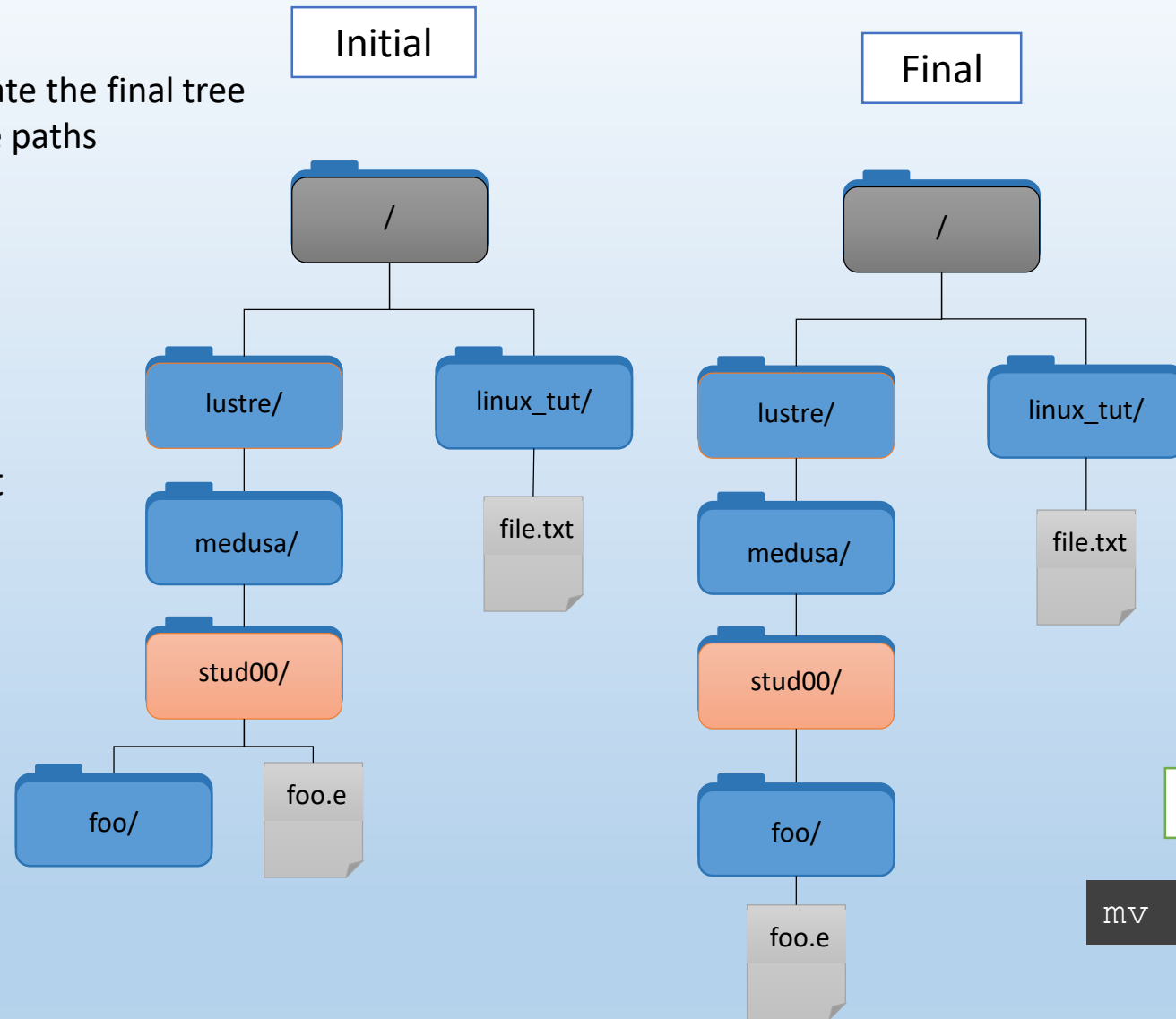
How does it work?

You will be given:

1. An initial path tree
2. A final path tree

Your mission:

- Write the commands that make the change using relative paths

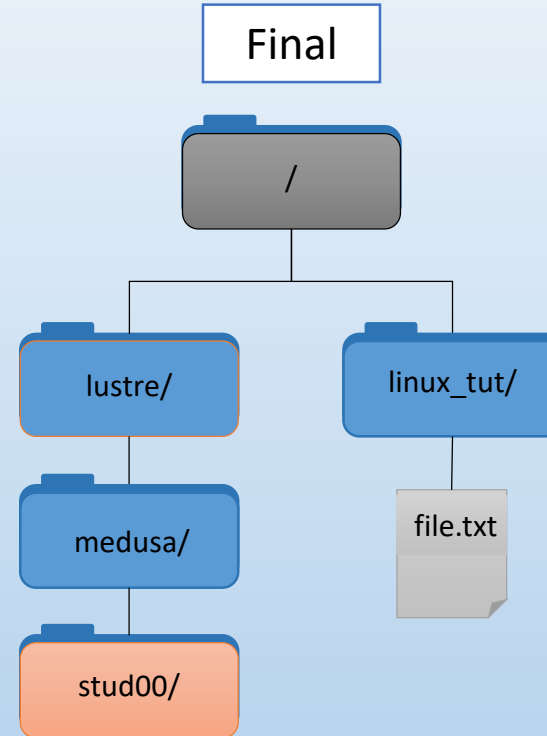
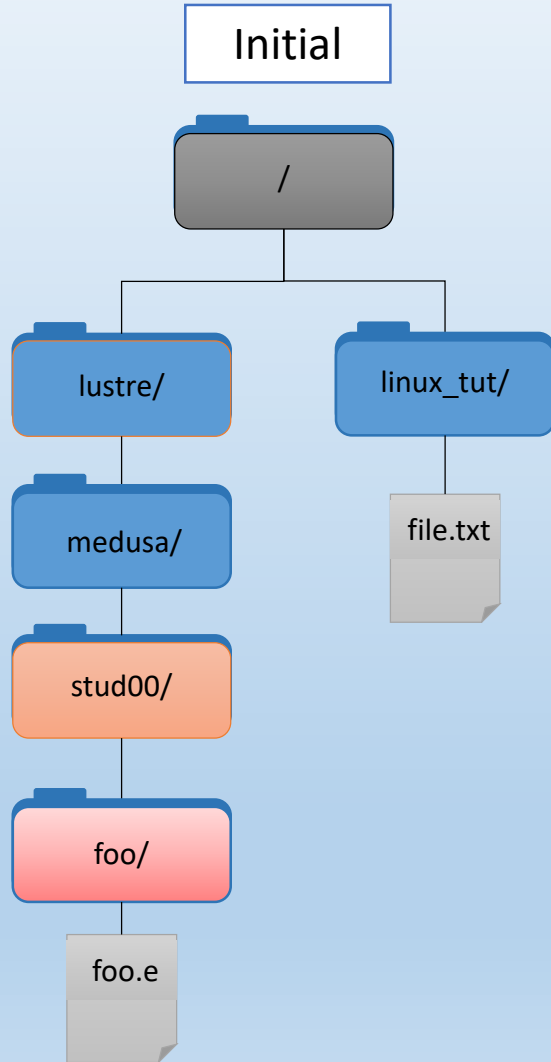


Command

```
mv foo.e foo/
```

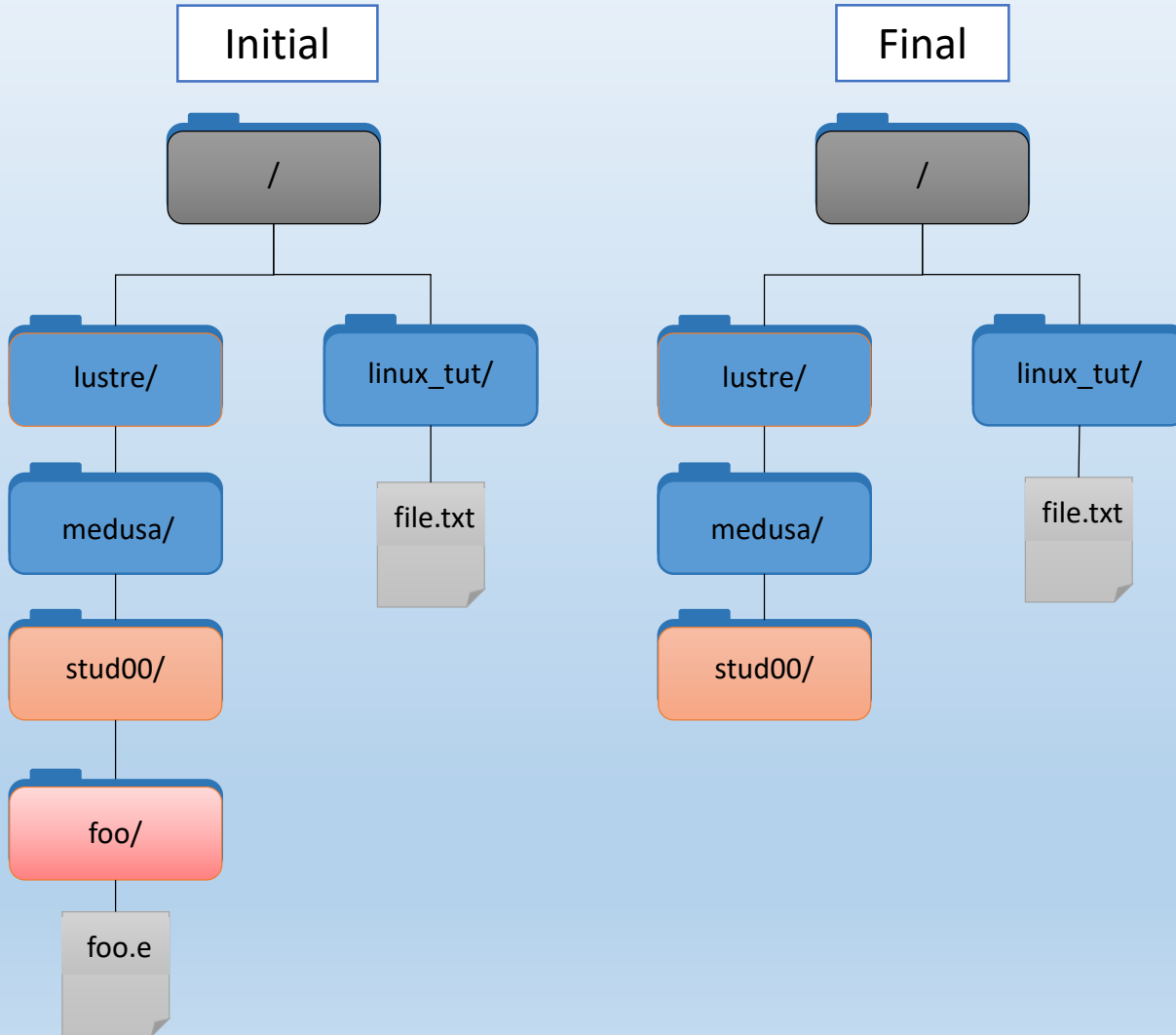
Special relativity challenge:

Write the Shell commands that generate the final tree from the initial tree using only relative paths



Special relativity challenge:

Write the Shell commands that generate the final tree from the initial tree using only relative paths

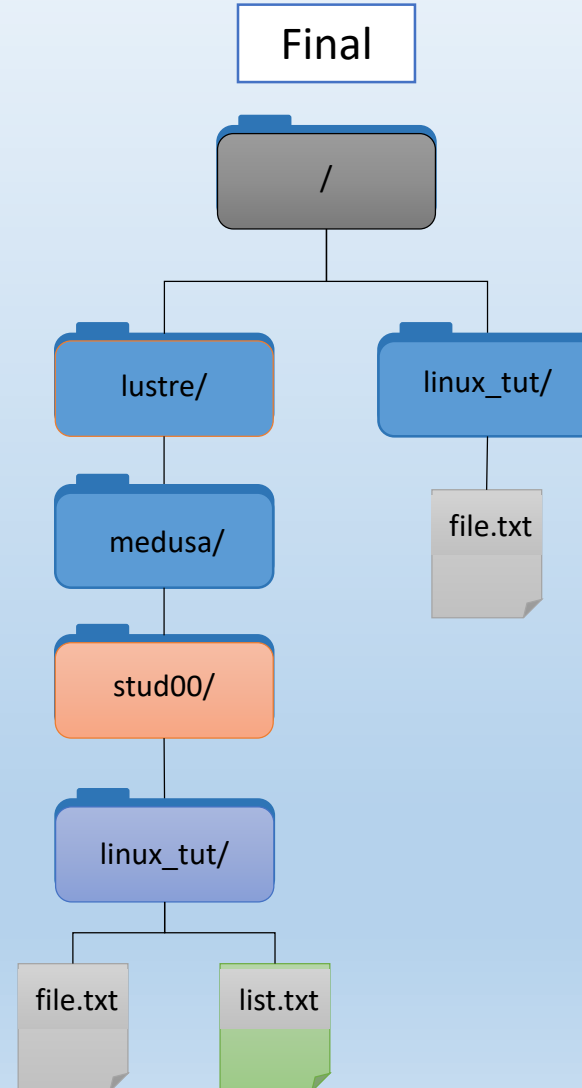
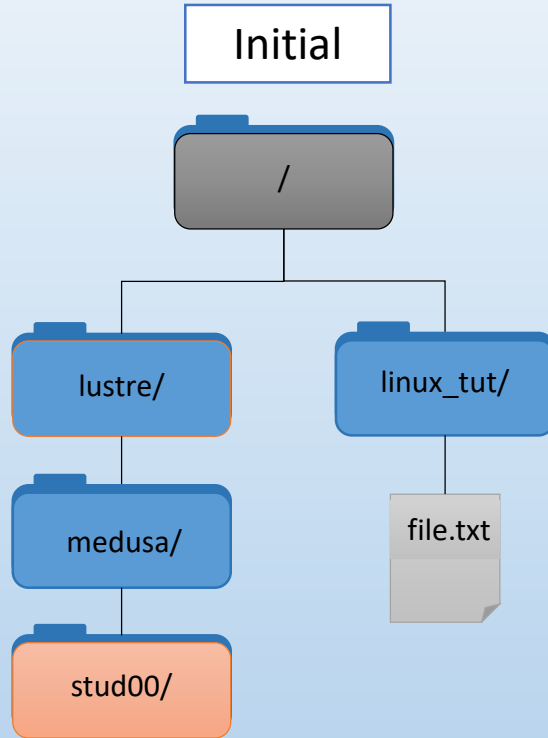


Commands

```
rm -r foo
```

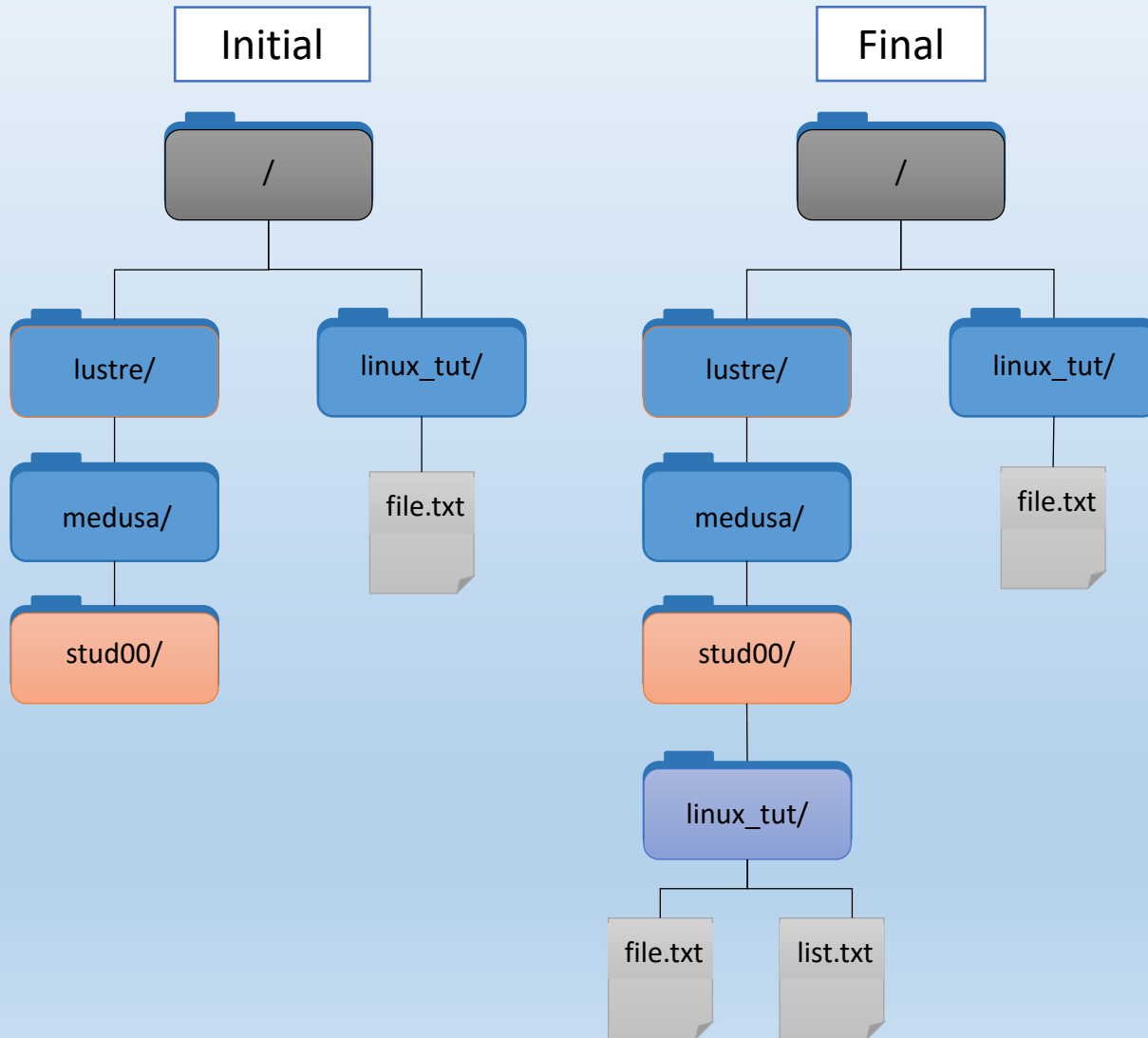
Special relativity challenge:

Write the Shell commands that generate the final tree from the initial tree using only relative paths



Special relativity challenge:

Write the Shell commands that generate the final tree from the initial tree using only relative paths



Commands

```
cp -r ../../../../linux_tut/ .  
nano linux_tut/list.txt
```

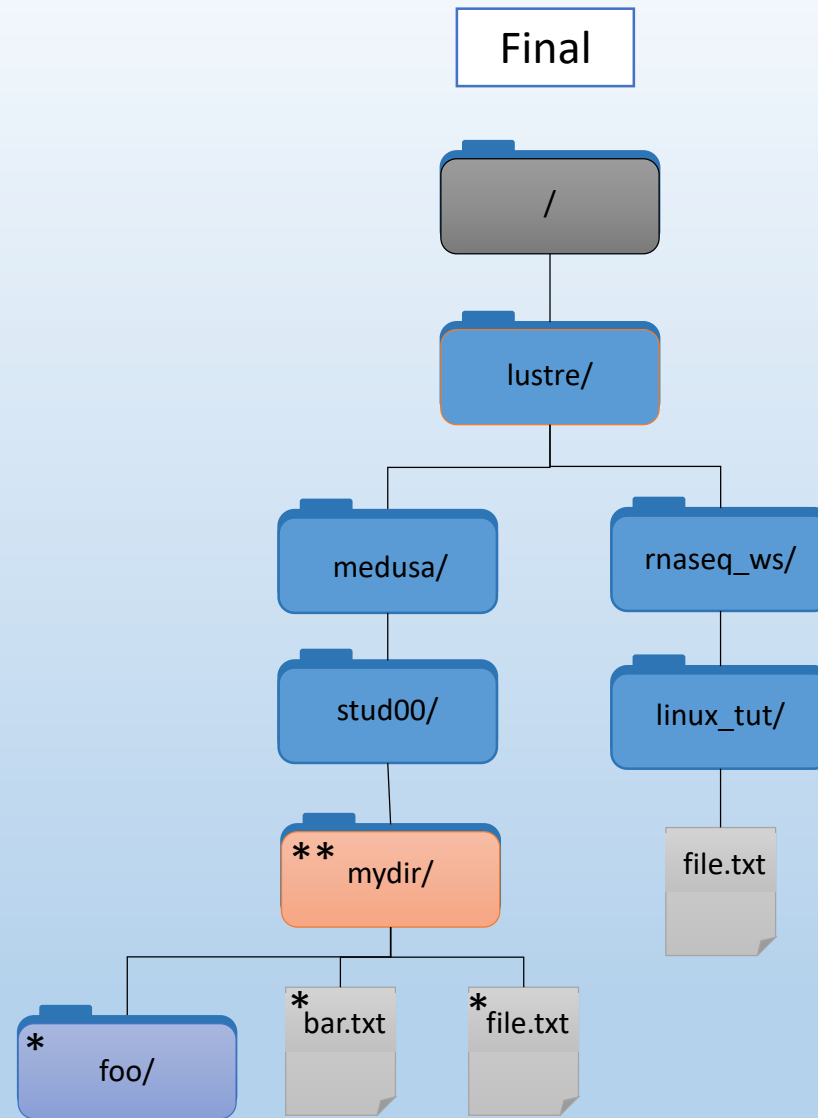
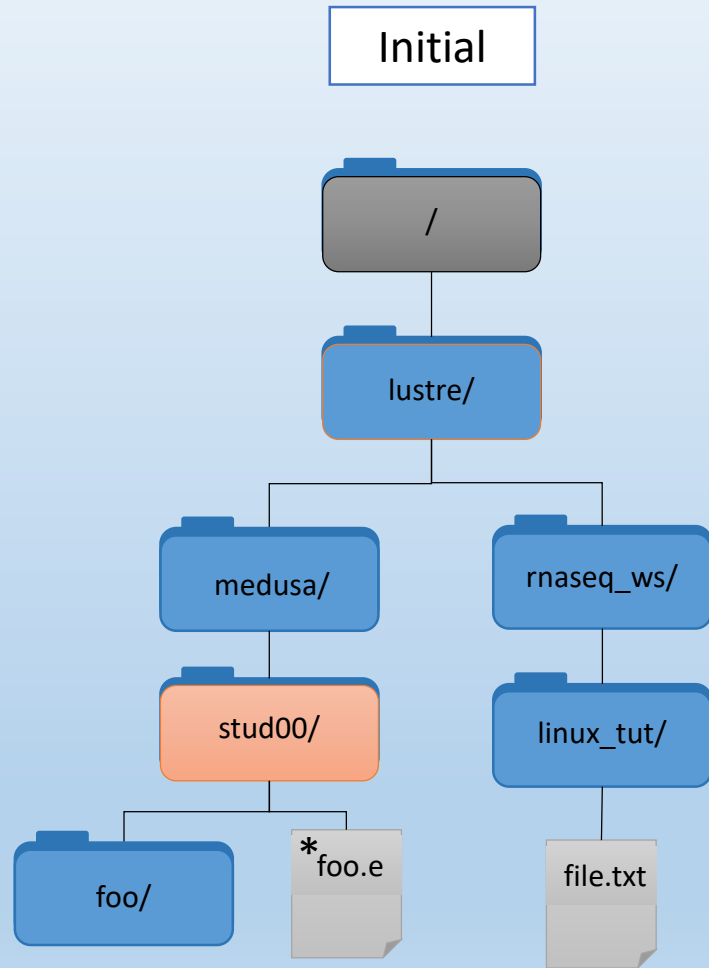
Another option is

```
mkdir linux_tut  
cp ../../../../linux_tut/file.txt linux_tut  
nano linux_tut/list.txt
```

Final challenge

Final challenge:

Write Shell commands that create the final tree from the initial tree



Final challenge:

Describe any differences you can find

Option 1

```
mkdir mydir
rm foo.e
cd mydir
mv ../foo .
nano bar.txt
cp /lustre/rnaseq_ws/linux_tut/file.txt .
```

Option 2

```
rm foo.e
mkdir mydir
mv foo/ mydir
cp ../../rnaseq_ws/linux_tut/file.txt mydir/
cd mydir
nano bar.txt
```

THANK YOU



ESCAPE

He finally did it!