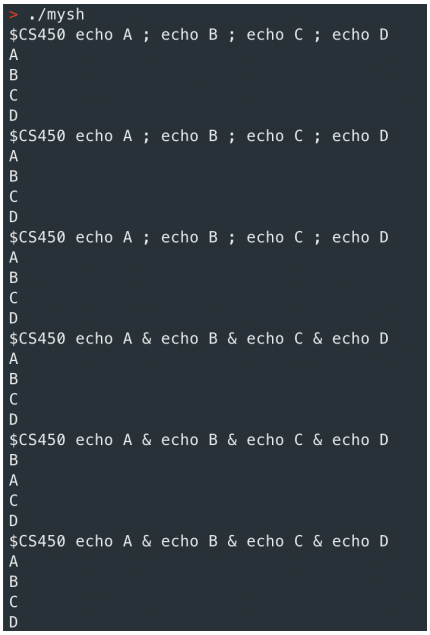
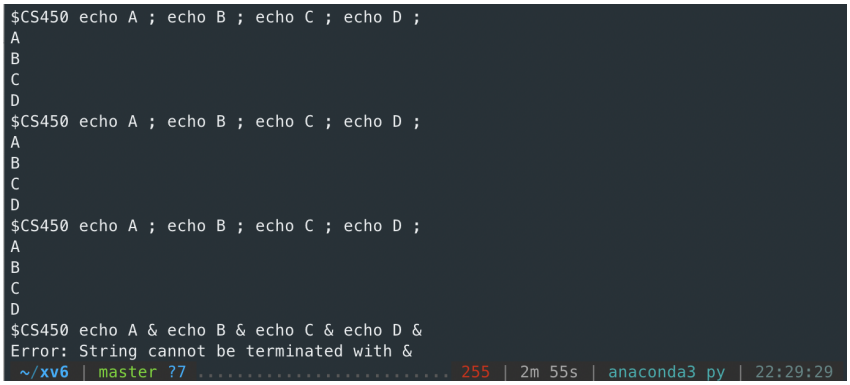
1) The test data below is of a good quality because it tests each of the ‘;’ and ‘&’ commands at least 3 times, which means it has repeatability



2) The test data below is of good quality because it shows what happens when the execution ends with ‘;’ or ‘&’. Each of the programs also have been tried at least 3 times to create repeatability. Note the program raises an error, when there is a ‘&’ at the end of the execution.



3) The test case below is of good quality because it shows how a program acts when there is a sequential execution sign ‘;’ that splits two parallel executions. Again, the program has been tested at least 3 times to create a repeatability.

```
$CS450 echo A & echo B ; echo C & echo D
A
B
C
D
$CS450 echo A & echo B ; echo C & echo D
A
B
D
C
$CS450 echo A & echo B ; echo C & echo D
A
B
C
D
```

4) The test case below is of good quality because it shows how a program acts when there is a parallel execution between the two sequential executions. Again, the program has been tested at least 3 times to create a repeatability.

```
$CS450 echo A ; echo B & echo C & echo D ; echo E
A
C
B
D
E
$CS450 echo A ; echo B & echo C & echo D ; echo E
A
B
C
D
E
$CS450 echo A ; echo B & echo C & echo D ; echo E
A
B
C
D
E
```

5) The test case below is of good quality because it shows that a program treats signs, other than ‘;’ and ‘&’ as characters of the first (or not) program, until it hits command signs ‘&’ or ;‘’. Each has been tested at least 3 times

```
$CS450 echo A - echo B & echo C ; echo D
A - echo B
C
D
$CS450 echo A = echo B & echo C ; echo D
C
A = echo B
D
$CS450 echo A ! echo B & echo C ; echo D
A ! echo B
C
D
```

```
$CS450 echo A * echo B ; echo C & echo D
A * echo B
C
D
$CS450 echo A + echo B ; echo C & echo D
A + echo B
D
C
$CS450 echo A # echo B ; echo C & echo D
A # echo B
C
D
```