CITS2003/CITS4407 In Semester Test 2023

Q1. Given the file name pattern \*a?\*t\* which of the following file names match that pattern **(2 marks each**):

atga Match Does Not Match

gctaat Match Does Not Match

gatt Match Does Not Match

atgata Match Does Not Match

Q2. Which of these statements is False:

* Free-to-use software is open-source
* Open-Source software is free to use
* Open-Source software is free to modify
* You are free to incorporate open-source software in other code (with due acknowledgement) **(2 marks)**

Q3. You are using your computer when you notice that is working hard on something; the fan is running all the time. A bit later, responses to typing have slowed markedly. In short, it is likely that a process has run off the rails.

1. Which Linux command can you use to identify the rogue process?

*top, htop, ps (ps not as good but info can be found there, so accept) 2 Marks*

1. Which Linux command can you use to stop the process that is causing the problem, but not other processes *kill, pkill 2marks*

Q4. Assume that the file a.sh contains the Shell script:

!/usr/bin/env bash[

size=$(ls -s $1 | cut -d' ' -f 1)

if test $size -ge 10000

then

echo $1

fi

Also assume that a.sh is called in the following way:

find ~ -name '\*.gz' -exec a.sh '{}' \;

It may be helpful for you to know that the Unix command ls -s on some file reports the size of the file (in 512 byte blocks), followed by the file name.

What does the call to find, including the call to a.sh do (in overview, not line by line)? **(4 marks)**

*Starting from the user’s home directory (home is sufficient) (1 mark), find all files whose names end in gz (ie gzip files) (1 mark) and report those which are at least 10,000 blocks in size (2 marks)*

Q5. What is antibugging and why is it important for programs to have it. **(4 marks)**

Antibugging is the addition of tests, typically near the start of a program, which ensure that that data coming from the user is consistent with what is expected. It is important because otherwise, nonsense results may be computed from absent, out of range or otherwise problematic data.

*Marks: 2 marks for providing a definition; 2 marks for saying why it is important*

Q6. Write a complete, runnable Bash program that, given a text file, will print to standard output every second line, starting with the first line (then the third line etc). **Please only use Bash commands**. (Hint: you do not need arithmetic for this; just use a variable that is given, in rotation, one of two values. However, if you wish to use Bash arithmetic, that is also fine.) Make sure to handle errors appropriately **(8 marks)**

#!/usr/bin/env bash

if [[ ! -s $1 ]]

then

echo "The file $1 does not exist" > /dev/stderr # redirection not important

exit 1 #exit important, status value is not important

fi

odd=True

IFS=" # Don’t worry if this is not present

"

for line in $(< $1) # while .. read also fine

do

if [[ $odd = True ]]

then

echo $line

odd=False

else

odd=True

fi

done

*1 mark for #! Line, 1 mark for file check, 2 marks for while loop across text file lines, 2 marks for if within loop, 4 marks for if test, etc that allows only odd numbered lines to be printed. The exit is important, but not the actual value.*

Q7**.** Write a complete, runnable Bash program called col\_count which, given the name of a single-tab-separated plain-text datafile, and a column number (from 1) – in that order – returns the item from the selected column that has the highest number of occurrences. (If there are several, any one of those with largest counts is fine.) Make sure to handle errors appropriately, but in particular include a test that the requested column number is within the range column numbers available in the file. You can assume that the first argument, if present, will be a string, and the second argument, if present, will be an integer.

For example, if datafile contains:

a a a

b a b

c b b

a c b

and the query is: col\_count datafile 3

the program should report: 3 b, or just b

The framework has to be a Bash script, but it can call other Unix tools. **(10 marks)**

#!/usr/bin/env bash

if [[ $# -ne 2 ]]

then

echo "Expecting 2 arguments <tsv text file> <col number>" > /dev/stderr

exit 1

fi

if [[ ! -s $1 ]]

then

echo "$0: the nominated file $1 does not exist or has zero length" > /dev/stderr

exit 1

fi

col\_count=$(head -1 $1 | tr ' ' '\012' | wc -l)

if test $2 -gt $col\_count -o $2 -lt 1

then

echo "Column number is greater than the number of columns or less than 1" > /dev/stderr

exit 1

fi

cut -d ' ' -f $2 < $1 | sort | uniq -c | sort -k 1nr | head -1

*Marks: 2 marks for other antibugging; 4 marks for column count test; 4 marks for extracting, sorting, etc the requested column*