

**Midterm 1**

Term: Fall 2018 (Sep4-Dec4)

Student ID Information

Last name:	First name:
Student ID #:	CS.Dal.Ca userid:

Course ID:	CSCI 2132	Grade Table	
		Question	Score
Course Title:	Software Development	1	/10
Instructor:	Vlado Keselj	2	/12
Date of Exam:	27 Sep 2018	3	/11
Time Period:	Start: 19:35 End: 20:25	4	/12
Duration of Exam:	50 minutes	5	/10
		Σ	/55
Exam Location:			
Number of Exam Pages: 8 pages (including this cover sheet)			
Exam Type:	Closed Book		
Additional Materials Allowed:	One letter-format paper (8.5" x 11") with anything written or printed on it (both sides). No textbooks, computers, calculators, or other aids are allowed.		

1. **(10 points) True-false questions:** 2 points each. Justification is not necessary, but brief justification may be helpful if correct.
 - a) (2 points) When we use PuTTY to login to a server, the OS kernel responds with a prompt string and waits for our command.
 - b) (2 points) Given a textual file `a.txt` The following command is an example of a pipeline command: `a.txt > sort > b.txt`
 - c) (2 points) If a file has the following permissions `-----r--`, the owner of the file cannot read it. (The order of permissions is the standard as in the `ls -l` command.)
 - d) (2 points) Two different directories cannot share the same inode number.
 - e) (2 points) If we use the C statement `scanf("%d", &n);` and enter the word “zero” in the input, an exception is thrown.

2. (12 points) Multiple-choice. No justification necessary. Circle the *single* best answer.

a) (3 points) If we run the following three commands:

'chmod 743 a.txt', 'chmod 632 a.txt', and 'chmod a+x a.txt'
then the final permissions of the file 'a.txt' will be as follows:

- A. `rw--wx-wx`
 - B. `rwx-wx-wx`
 - C. `rw--wx-x-`
 - D. `rw--w--w-`
-

b) (3 points) After executing the following two C statements:

'`x = 3.0; n = scanf("%lf", &x);`' on the following input '`-.01e1 2.2`', the values of int variable n and double variable x are:

- A. $n = 0$ and $x = 3.0$
 - B. $n = 1$ and $x = 3.0$
 - C. $n = 0$ and $x = -0.01$
 - D. $n = 1$ and $x = -0.1$
-

c) (3 points) A file named `courses.csv` contains courses taken by a group of students, where each line has a format: *student name, course number, term*. An example of a line from such file would be: John Smith, CSCI 2132, Fall 2018

Which command will print the number of different courses in the file?

- A. `cut -d "," -f 2 < courses.csv | sort | uniq | wc -l`
 - B. `sort < courses.csv | cut -d "," -f 4 | uniq | wc -l`
 - C. `cut -d "," -f 2 < courses.csv | wc -l | uniq | sort`
 - D. `cut -d "," -f 4 < courses.csv | sort | wc -l | uniq`
-

d) (3 points) You created a new file in your working copy, and you want to save this file in the SVN repository. The commands that you need to use for this task are:

- A. `svn checkout` and `svn commit`
- B. `svn add` and `svn commit`
- C. `svn update` and `svn commit`
- D. `svn resolved` and `svn commit`

3. (11 points) Give concise answers.

- a) (3 points) Give example of an absolute and a relative pathname. (Label which one is absolute and which one is relative.)
- b) (4 points) When do we use wildcards and when regular expressions? Show by example a difference between their notations.
- c) (4 points) Draw the directory structure created by the following commands and circle our current directory at the end.

```
mkdir dir ; cd dir
mkdir -p a/a a/b a/c b/a b/b b/c/a
cd b/c/a ; mkdir -p ../../b/c/c; cd ../a
```

4. (12 points) **Command line.** For each of the following questions write a single command line to perform the required task. You do not need to justify the answer. You are allowed to use only the following commands, with possible pipes: `cat`, `cut`, `echo`, `grep`, `egrep`, `ls`, `sort`, `uniq`, and `wc`.

a) (4 points) Print out a single integer that is the number of files in the directory `/home/joe/etc` whose names start with a lowercase letter, does not end with ‘c’ or ‘d’ or a digit, and are five characters long. An example of such filename would be `/home/joe/etc/abcdF`

Note that you are asked to print the number of such files.

b) (4 points) Print out all lines from a file named `file.txt`, that contain the word ‘printf’ or ‘scanf’, and also contain the word ‘return’. Make sure to match exact words, for example ‘returning’ should not be matched with ‘return’.

c) (4 points) You are given a file named `/etc/passwd`, which contains a list in the following format:

```
root:x:0:0:root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
user1:x:1000:1000:John Doe:/home/user1:/bin/tcsh
```

Each line contains seven fields separated by colon (‘:’). Write a command to sort the lines of the file by the fifth (5th) field, and then print only the first five fields of each line.

5. (10 points) C Program.

(10 points) Given three positive numbers a_1 , a_2 , and a_3 , the arithmetic mean (AM) and harmonic mean (HM) of these numbers are defined as:

$$AM = \frac{a_1 + a_2 + a_3}{3}, \quad HM = \frac{3}{\frac{1}{a_1} + \frac{1}{a_2} + \frac{1}{a_3}}$$

Write a C program that asks a user to enter three positive floating-point numbers, separated by semi-colons. The program then computes the arithmetic mean and harmonic mean of these three numbers, and prints the results as floating-point numbers, keeping **4 digits after the decimal point**.

You can make your own (reasonable) assumptions about how the program handles the input and output, as long as your program will work if the user enters the input values using the format ‘ a_1 ; a_2 ; a_3 ’, e.g. ‘0.5 ; 2.5 ; 7’. You can notice that user will use semi-colon between numbers and possibly some spaces.

You can assume that the user always correctly enters the positive numbers in the required format. No error handling is required. You are only allowed to use variables whose types are **float** or **double** in your program.

(extra blank page 1 provided)

(extra blank page 2 provided)