

CSCA20 Fall 2022 Final Exam  
 Duration — 2 hours and 50 minutes  
 Aids allowed: Printed Material Only,  
 No Electronic Devices

Student Number: \_\_\_\_\_

UTORid: \_\_\_\_\_

Family Name: \_\_\_\_\_

Given Name: \_\_\_\_\_

Read and follow all instructions on this page, and fill in all fields appropriately.

*Do **not** turn this page until you have received the signal to start.*

(Please fill out the identification section above, and read the instructions below.)

*Good Luck!*

This exam is double-sided, and consists of 6 questions on 14 page (including this one). When you receive the signal to start, please make sure that you have all pages.

- Read all instructions before completing any questions
- Do not remove any pages from the exam booklet.
- Write your details on the top of the first page.
- Write your name and indicate your tutorial on the back of the last page.
- If you use any space for rough work, indicate clearly what you want marked.
- Write as clearly and legibly as possible. No marks will be awarded to unreadable answers.

#### Competencies & Masteries Demonstrated

	User I/O		Variables		Selection		Loops		Files		Functions		Dicts		External	
	C	M	C	M	C	M	C	M	C	M	C	M	C	M	C	M
Q1																
Q2																
Q3																
Q4																
Q5																
Q6																
Total																

```
def gets_present(days_list):  
    #INPUT:  
    # a list of strings representing the behaviour of a child on each day  
    # of the year. Each entry in the list should be either "Naughty",  
    # "Nice", or "No Info"  
    #OUTPUT:  
    # returns True if the child gets a present. A child gets a present  
    # if the number of "Nice" entries in the list is equal to or greater  
    # than the number of "Naughty" entries
```

**Question 1.**

**Completion of this question demonstrates competency in loops and selection**

In the space below, write the code to complete the function on the opposite page

*[Use the space below for rough work. This page will not be marked unless you clearly indicate the part of your work that you want us to mark.]*

## Question 2.

**Completion of this question demonstrates competency in loops, files and functions, and allows for demonstration of mastery in functions**

Write a function called `get_behaviour_list` that takes the name of a file (not the already open file handle), and returns a list of strings. The first line of the file will contain the name of a child, and the next 365 lines will represent their behaviour on each day of the year (the lines will read either **Naughty**, **Nice**, or **No Info**). Your function should return a list of behaviours suitable for input into the `get_presents` function from Question 1.

To demonstrate mastery in functions, your function must be well documented with clear explanation of inputs and outputs as well as internal commenting.

Menu Options:

- N = New Child - Ask the user for the name of a child, and read in the child's current behaviour file
- U = Update Behaviours - Prompt the user for a day (1-365) and a behaviour ("Naughty" or "Nice") and update the child's behaviours list accordingly
- R = Report - Tell the user whether the child will be getting a present given their current behaviours list
- W = Write Behaviours File - Replace the child's existing behaviours file with a new one representing any updated behaviours

You may assume that all children have existing behaviour files named after them (e.g., Alice will have a behaviour file called `Alice.txt`).

### Question 3.

**Completion of this question demonstrates competency in User I/O, Variables, Loops, Selection, and Files, and allows for demonstration of mastery in User I/O**

Create a program for manipulating the behaviour files for children. For this question, you may assume you have working `gets_present` and `get_behaviour_list` functions even if you did not complete them in Questions 1 and 2.

Your code should give the user the options listed on the opposite page and perform the corresponding actions.

To demonstrate mastery in User I/O, your menu must recover from all bad input and provide helpful error messages.

Example CSV file

```
Name,Present,Nice Day Count,Naughty Day Count
Alice,Train,312,23
Bob,Teddy Bear,250,43
Carol,Train,100,103
```

Example Output

```
{'Alice': 'Train', 'Bob': 'Teddy Bear', 'Carol': 'Coal'}
```



## Question 4.

**Completion of this question demonstrates competency in files, dictionaries, and external tools**

Given a csv file formatted as on the opposite page, produce a dictionary mapping the name of a child to the present they receive. If the child has more naughty days than nice days, they should receive `coal`. Otherwise, they should receive the present they want. To receive competency in external tools, you must use the csv module to read the file.

## Sample Input File

```
5
3,French Hens
1,Partridge in a pear tree
2,Turtle Doves
5,Gold Rings
```

## Sample Output File

```
On day 1 of Christmas, my true love gave to me:
1 Partridge in a pear tree
On day 2 of Christmas, my true love gave to me:
2 Turtle Doves
1 Partridge in a pear tree
On day 3 of Christmas, my true love gave to me:
3 French Hens
2 Turtle Doves
1 Partridge in a pear tree
On day 4 of Christmas, my true love didn't give me anything
On day 5 of Christmas, my true love gave to me:
5 Gold Rings
3 French Hens
2 Turtle Doves
1 Partridge in a pear tree
```

**Question 5.**

**Completion of this question demonstrates mastery in variables, loops, selection, and files**

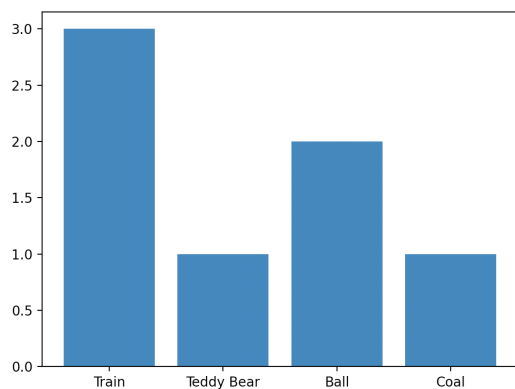
Given a file called `gifts.txt` formatted such that the first line is the number of days for the song, and the remaining lines are the day number (not in order) and gifts for that day. Write a file called `poem.txt`, where each day, the list of presents for that day and all previous days are given formatted as on the opposite page.

## Example Input Dictionary

```
{'Alice': 'Train', 'Bob': 'Teddy Bear', 'Carol': 'Train',  
'Dave': 'Ball', 'Edith': 'Train', 'Frank': 'Coal', 'Gertrude': 'Ball'}
```

## Example Output

```
{'Train': {'Carol', 'Edith', 'Alice'}, 'Teddy Bear': {'Bob'},  
 'Ball': {'Dave', 'Gertrude'}, 'Coal': {'Frank'}}
```



**Question 6.**

**Completion of this question demonstrates mastery in dictionaries and external tools**

Given a dictionary mapping children's names to the toys they have requested, produce a dictionary mapping toys to the set of children that will receive that toy. Then use that dictionary plot a bar chart of the number of children who got each present.

