

Lab 2.2 (Deadline Friday 28 January 23:59)

- Upload your code to [Einstein](#) to have it verified.

Best student version 1

- Write a program called `best_v1_022.py` that takes a filename as a single command line argument. The file contains a list of student marks where each line consists of a mark followed by a student name.
- The program should print out the name of the student who achieved the highest mark as well as the mark itself. Where there is a joint top mark the program should print the name associated with the first one encountered, for example:

```
$ cat best_v1_input_00_022.txt
64 Mary Ryan
89 Michael Murphy
22 Pepe
78 Jenny Smith
57 Patrick James McMahon
89 John Kelly
22 Pepe
74 John C. Reilly
```

```
$ python3 best_v1_022.py best_v1_input_00_022.txt
Best student: Michael Murphy
Best mark: 89
```

- If the filename specified on the command line cannot be opened the program must make appropriate use of exception handling to display an error message and exit gracefully. For example:

```
$ python3 best_v1_022.py best_v1_input_00_022.txtt
The file best_v1_input_00_022.txtt could not be opened.
```

Best student version 2

- Extend the exception handling in the above program such that the program exits gracefully if any of the marks in the input file are not integers. Call the new program `best_v2_022.py`. For example:

```
$ cat best_v2_input_00_022.txt
64 Mary Ryan
89 Michael Murphy
22 Pepe
78 Jenny Smith
5a Patrick James McMahon
89 John Kelly
22 Pepe
```

```
74 John C. Reilly
90 Penelope Pitstop
```

```
$ python3 best_v2_022.py best_v2_input_00_022.txt
Invalid mark 5a encountered. Exiting.
```

Best student version 3

- Extend the exception handling in the above program such that rather than exiting on encountering a non-integer mark the program simply ignores it and continues to process all remaining marks. Call the new program `best_v3_022.py`. For example:

```
$ cat best_v3_input_00_022.txt
64 Mary Ryan
89 Michael Murphy
22 Pepe
78 Jenny Smith
5a Patrick James McMahon
89 John Kelly
22 Pepe
74 John C. Reilly
90 Penelope Pitstop
```

```
$ python3 best_v3_022.py best_v3_input_00_022.txt
Invalid mark 5a encountered. Skipping.
Best student: Penelope Pitstop
Best mark: 90
```

Best students

- The above programs print out a single name even when multiple students share the highest mark. Write a program called `bests_022.py` that prints out the names of all students who achieved the highest mark.
- Student names should be comma-separated and must be printed in the order in which they occur in the input file. For example:

```
$ cat bests_input_00_022.txt
64 Mary Ryan
89 Michael Murphy
22 Pepe
78 Jenny Smith
57 Patrick James McMahon
89 John Kelly
22 Pepe
74 John C. Reilly
```

```
$ python3 bests_022.py bests_input_00_022.txt
Best student(s): Michael Murphy, John Kelly
Best mark: 89
```

Two files

- Write a program called `twofiles_022.py` that takes two filenames as command line arguments. Each file contains a list of integers (one per line) and each file contains the same number of integers.
- The program should output a list of integers, one per line, where integers are taken alternately from each file. For example:

```
$ cat twofiles_input_a_00_022.txt
1
3
5
7
```

```
$ cat twofiles_input_b_00_022.txt
2
4
6
8
```

```
$ python3 twofiles_022.py twofiles_input_a_00_022.txt twofiles_input_b_00_022.txt
1
2
3
4
5
6
7
8
```

Exceptions puzzles

- Without running them, write out the output of each of the following programs (run each afterwards to check your answer):

```
#!/usr/bin/env python3

def callme(x):
    y = None
    try:
        y = 1//x
    except ZeroDivisionError:
        print('In exception handler')
    else:
        print('Reached end of try')
    finally:
        print('Exiting the function')
    return y

def main():
    print(callme(1))
    print(callme(0))

if __name__ == '__main__':
    main()
```

```
#!/usr/bin/env python3

def callme(x):
    try:
        y = 1//x
    except ZeroDivisionError:
        print('In exception handler')
    else:
        print('Reached end of try')
    finally:
        print('Exiting the function')
    return y

def main():
    print(callme(1))
    print(callme(0))

if __name__ == '__main__':
    main()
```

```
#!/usr/bin/env python3

def callme(x):
    try:
        y = 1//x
        return y
    except ZeroDivisionError:
        print('In exception handler')
    else:
        print('Reached end of try')
    finally:
        print('Exiting the function')

def main():
    print(callme(1))
    print(callme(0))

if __name__ == '__main__':
    main()
```

- Note there is no marker on Einstein for these puzzles so you do not need to upload anything.