

Bucketlist exercises

- Upload your solution to [Einstein](#) before the deadline.

Exercise 1 : Word calculator (Deadline Sunday 13 March 23:59)

- Jimmy has invented a new kind of calculator that works with words rather than numbers.
- Input is read from `stdin` and consists of up to 1000 commands, one per line.
- Each command is a *definition*, a *calculation* or a *clear*.
- All tokens within a command are separated by single spaces.
- A definition has the format `def x y` where `x` is a variable name and `y` is an integer in the range `[-1000, 1000]`.
- Existing definitions are replaced by new ones i.e. if `x` has been defined previously, defining `x` again erases its old definition.
- Variable names consist of 1-30 lowercase characters.
- No two variables are ever defined to have the same value at the same time.
- The `clear` command erases all existing variable definitions.
- A calculation command starts with the word `calc`, and is followed by one or more variable names separated by addition or subtraction operators.
- The end of a calculation command is an equals sign.
- Write a program called `wordcalc_122.py` that implements Jimmy's calculator.
- The program should produce no output for definitions, but for calculations it should output the value of the calculation.
- Where there is no word for the result, or some word in a calculation has not been defined, then the output should be `unknown`. (The word `unknown` is never used as a variable name.)
- Your solution may only import content from the `sys` module.
- Your solution may not use the `eval()` function.
- For example:

```
$ cat wordcalc_stdin_00_122.txt
def foo 3
calc foo + bar =
def bar 7
def programming 10
calc foo + bar =
def is 4
def fun 8
calc programming - is + fun =
def fun 1
calc programming - is + fun =
clear
```

```
$ python3 wordcalc_122.py < wordcalc_stdin_00_122.txt
foo + bar = unknown
foo + bar = programming
programming - is + fun = unknown
programming - is + fun = bar
```

Exercise 2 : Vertical sorting (Deadline Friday 1 April 23:59)

- Write a program called `vertical_122.py` that reads, sorts and prints vertical words.
- Input is read from `stdin` and consists of an arbitrary number of lines of text.
- Each line contains the same number of characters.
- Characters are lower-case or upper-case letters.
- Consider the block of text as a sequence of columns of characters.
- Output is the same set of columns sorted in increasing alphabetical order (ignoring case).
- Where columns are identical (after case conversion) they should be output in the order they were read from `stdin`.
- Your solution may only import content from the `sys` module.
- Your solution may not use the `eval()` function.
- For example:

```
$ cat vertical_stdin_00_122.txt
ThatCc
IabiLl
Mpomoa
Epuecc
Sytskk
```

```
$ python3 vertical_122.py < vertical_stdin_00_122.txt
acChTt
bllLaIi
oaopMm
uccpEe
tkkySs
```

- For example:

```
$ cat vertical_stdin_01_122.txt
pdDP1
qeEQm
rfFrn
```

```
$ python3 vertical_122.py < vertical_stdin_01_122.txt
dDlpP
eEmqQ
fFnrr
```

Frequently asked questions

- **Q.** How are bucketlist exercises marked?
- **A.** See [here](#).