



# [CS 11 25.1] Lab 2b – Command Line

## Problem Statement

[Submit solution](#)

You have just written a *command-line program* for your CS 11 class! For our purposes, a command-line program consists of the following parts:

- The *program name* to identify your command-line program.
- A number of *option-value pairs* to control the functionality of the program.

As an example, suppose you wrote a command-line program to get the last  $k$  lines of a file. If you wanted, say, the last  $k = 11$  lines, you would run

```
tail -n 11
```

[Copy](#)**✓ Points:** 225 (partial)**⌚ Time limit:** 3.0s**⌘ Memory limit:** 1G**➤ Problem type****▼ Allowed languages**

py3

Try this out in your lab computer's terminal! Include the file you want to get the trailing lines of (e.g., `tail -n 11 main.py`).

Here, `tail` is the program name, and `(-n, 11)` is an option-value pair for the `tail` program.

Given a command that is invoked, can you identify all of its option-value pairs?

## Task Details

Your task is to implement a function named `get_options`, which should have the following *signature*:

```
def get_options(*command):
```

[Copy](#)

The above says that it has a variable number of arguments given as `command`.

The function must return a `frozenset` of pairs (tuple of length 2), where each pair contains two strings (`str`) denoting an option and its value.

## Restrictions

- The following symbols can now be used: `min`, `max`, `sum`, `range`, `all`, `any`.
- recursion is *disallowed*.
- comprehensions are allowed.
- at most 6 functions can be defined.
- Your source code must have at most 600 bytes.

## Examples

### Example 1 Function Call

```
get_options("tail", "-n", "11")
```

[Copy](#)

### Example 1 Return Value

```
frozenset((("-n", "11"),))
```

[Copy](#)

## Constraints

- The function `get_options` will be called at most 20 times.
- `command` follows the format outlined in the problem statement; that is,
  - It starts with the command name, and
  - The command name is followed by option-value pairs.
- The command name is a nonempty string of lowercase English letters with length at most 8.
- Each option is a nonempty string that consists of a dash (-) followed by a single lowercase English letter.
- Each value is a nonempty string with length at most 20 that consists of digits and/or uppercase/lowercase English letters.
- There are at most 20 options.

## Scoring

**Note:** New tests may be added and all submissions may be rejudged at a later time. (All future tests will satisfy the constraints.)

- You get 25 ❤ points if you solve all test cases where:
  - There is at most one option.
- You get 200 🎯 points if you solve all test cases.

## Clarifications

[Report an issue](#)

No clarifications have been made at this time.