

NO V0w3ls

Learning Goals

- Practice using strings
- Practice using command-line arguments
- Write a program entirely from scratch

Background

If you've been on the internet, you might have seen "leetspeak" (or "l33tsp36k" for our purposes!), which involves the substitution of symbols for alphabetical characters, where those symbols somewhat resemble their alphabetical counterparts. In this lab, you'll write a program to replace certain vowels with digits!

Up until now, you've frequently written programs for which you've been provided distribution code. You'll notice when downloading the "distro" for this problem, you start with nothing more than a couple of commonly used libraries and an empty `main` function. In this problem, you will convert a word, which you will input at the command line, to a corresponding word with numbers replacing vowels.

- Hints
 - Do note that the `main` function in the distribution code uses `argc` and `argv`. Be sure to use these!
 - Be sure to check for the correct number of command-line arguments before doing anything else in your `main` function.
 - Andrew's addition: just do it without distribution code, you know what you're doing.

Demo

```
no-vowels/ $ █
```

Getting Started

1. Log into cs50.dev using your GitHub account.
2. Click inside the terminal window and execute `cd`.
3. At the `$` prompt, type `mkdir no-vowels`
4. Now execute `cd no-vowels`
5. ~~Then copy and paste `wget https://cdn.cs50.net/2022/fall/labs/2/no_vowels.c` into your terminal to download this lab's distribution code.~~ Just create a `no-vowels.c` file and get writing!

Implementation Details

- Implement your program in a file called `no-vowels.c` in a directory called `no-vowels`.
- Your program must accept a single command-line argument, which will be the word that you want to convert.
- If your program is executed without any command-line arguments or with more than one command-line argument, your program should print an error message of your choice (with `printf`) and `return` from `main` a value of 1 (which tends to signify an error) immediately.
- Your program must contain a function called `replace` which takes a `string` input and returns a `string` output.
- This function will change the following vowels to numbers: `a` becomes `6`, `e` becomes `3`, `i` becomes `1`, `o` becomes `0` and `u` does not change.
- The input parameter for the `replace` function will be `argv[1]` and the return value is the converted word.
- The main function will then print the converted word, followed by `\n`.
- You may want to try using the `switch statement` in your `replace` function.

Thought Question

Why might you want to use command-line arguments rather than `get_string`, `get_int`, etc?

How to Test Your Code

Your program should behave per the examples below.

```
no-vowels/ $ ./no-vowels
Usage: ./no-vowels word
```

```
no-vowels/ $ ./no-vowels hello
h3110
```

```
no-vowels/ $ ./no-vowels pseudocode
ps3ud0c0d3
```

You can check your code using `check50`, a program that CS50 will use to test your code when you submit, by typing in the following at the `$` prompt. But be sure to test it yourself as well!

```
check50 cs50/labs/2022/fall/no-vowels
```

Green smilies mean your program has passed a test! Red frownies will indicate your program output something unexpected. Visit the URL that `check50` outputs to see the input `check50` handed to your program, what output it expected, and what output your program actually gave.

To evaluate that the style of your code (indentations and spacing) is correct, type in the following at the \$ prompt.

```
style50 no-vowels.c
```

How to Submit

No need to submit! This is an optional practice problem completed with your lab.