

Screenshot for part one. The only meaningful difference between my interpretation and the python script is I elected to convert the string input into an array as I pass it into the palindrome detecting function because I felt like dealing with indexes more than I felt like dealing with the substring function.

The screenshot shows a terminal window with the following content:

```
palindrome.pl
palindrome.pl
1 sub palindrome {
2     if (@_ <= 1){
3         return "1";
4     }
5     else {
6         if (lc($_[0]) eq lc($_[-1])){
7             return palindrome(@_[1..$#-1]);
8         }
9         else{
10            return "0";
11        }
12    }
13 }
14
15 print palindrome split("//,@ARGV[0]); print "\n"
```

Below the code, the terminal shows the execution of the script with various inputs:

- rthompson@DESKTOP-EAHC8Q8:~/arizona/hlt538-computational_linguistics/hw3\$ perl palindrome.pl d
- 1
- rthompson@DESKTOP-EAHC8Q8:~/arizona/hlt538-computational_linguistics/hw3\$ perl palindrome.pl dad
- 1
- rthompson@DESKTOP-EAHC8Q8:~/arizona/hlt538-computational_linguistics/hw3\$ perl palindrome.pl dad da
- 1
- rthompson@DESKTOP-EAHC8Q8:~/arizona/hlt538-computational_linguistics/hw3\$ perl palindrome.pl cad
- 0
- rthompson@DESKTOP-EAHC8Q8:~/arizona/hlt538-computational_linguistics/hw3\$ perl palindrome.pl dada
- 0
- rthompson@DESKTOP-EAHC8Q8:~/arizona/hlt538-computational_linguistics/hw3\$ perl palindrome.pl daad
- 1
- rthompson@DESKTOP-EAHC8Q8:~/arizona/hlt538-computational_linguistics/hw3\$ perl palindrome.pl racecar
- 1
- rthompson@DESKTOP-EAHC8Q8:~/arizona/hlt538-computational_linguistics/hw3\$ []

Perl code”

sub palindrome {

```

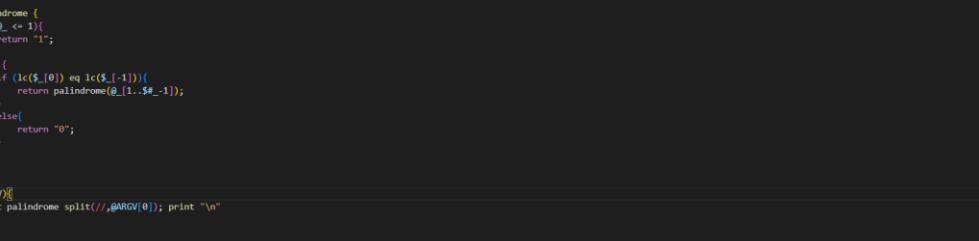
if (@_ <= 1){
    return "1";
}
else {
    if (lc($_[0]) eq lc($_[-1])){
        return palindrome(@_[1..$#_ -1]);
    }
    else{
        return "0";
    }
}

```

```
print palindrome split(//,@ARGV[0]); print "\n"
```

6

Part 2 screenshot. The only code modified is in the print statements at the bottom.
Checked the @ARGV array as a scalar value within an if statement to see if it's empty or not



The screenshot shows a code editor window in VS Code with the following details:

- Title Bar:** The title bar displays "File Edit Selection View ... < > hw3 [WSL: Debian] ⚡".
- Left Sidebar:** A sidebar on the left contains icons for file operations like Open, Save, Find, and others.
- Code Editor:** The main area shows a Perl script named "palindrome.pl". The code defines a subroutine "palindrome" that checks if a string is a palindrome. It handles edge cases where the string length is 1 or 0. For strings longer than 1 character, it compares the first and last characters and then recursively calls itself on the remaining substring. If no arguments are provided, it prints usage instructions.
- Bottom Navigation:** The bottom navigation bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS.
- Terminal:** The terminal tab shows command-line interactions with the script. It starts with a help message, followed by three runs of the script with different inputs ("1", "1234321", and "123456789").

Part 3:

The screenshot shows a code editor with a Python file named `hw3-template-palindrome.py`. The code defines a function `palindrome(a)` that checks if a string is a palindrome. It handles edge cases where the string length is 1 or 0. For strings longer than one character, it recursively checks the substring from index 1 to -1. The script also includes a check for command-line arguments and prints the result to the console.

```
#HW3 template: palindrome.py
#Edited to handle calling without arguments smoothly
import sys
def palindrome(a):
    if len(a) <= 1:
        return True
    else:
        if a[0] == a[-1]:
            return palindrome(a[1:-1])
        else:
            return False
if __name__=="__main__":
    print(palindrome(sys.argv[1]))
else:
    print("Usage: python3 palindrome.py <string to evaluate>")
```

The terminal below shows the execution of the script with different inputs:

- Running the script without arguments results in usage instructions.
- Running with the argument "dad" returns `True`.
- Running with the argument "False" returns `False`.