Jasque Saydyk

Professor Vanderberg, Swenson, and Trice

CS 126L Section 2

28 January 2017

Lab 01 - MadLib

1. Problem Statement

The important features of this problem is the program needs to ask for input from the user, then accept and store the given input, then use the given input to insert it in a predefined paragraph of text that makes some amount of sense.

Requirements

- Paragraph with needed components and words noted
- Variables to store string inputs
- Asking the user for input
- Display paragraph with noted components and words replaced with input

2. Planning

While we can get fancy with this project and implement the input and output as separate functions in a separate class, then just run them in a main class, we decided not to do this. This is because the main advantage this would bring is helping us narrow down the source of an issue if we were to have one, and this problem is simple enough to where we don't expect we be of any use at all. Therefore we will implement this program in a procedural style in the main method of the program.

We could also make use of lists and loops to potentially cut down on the amount of code that needs to be written, however the problem isn't complex enough to warrant this treatment, and it may make the code for the output less readable.

As for error checking on the input, we decided that it wasn't necessary as the worst the user will be able to do is make their paragraph of text make no sense, and the simple act of ensuring the user inputs a word that exists increases the scope of the project to an unreasonable amount.

For gathering the input from the user, we will use the built in Input() function in Python, which will allow the user to use spaces and other common keyboard strokes in their answer without bugging the program. The only other issue that need to be addressed is the usage of + to concatenate the paragraph with the input. Python, unlike other languages, has a reasonably fast

algorithm for the + to concatenate strings, and we would only consider switching to another method if we had a truly large number of strings to concatenate and we noticed the lag.

3. Implementation and Testing

Implementation of this plan was very straight-forward. After creating a list of variables, we then made an input statement for each variable. After giving each variable a String value, we then inserted it into the paragraph, concatenating it all together. As for testing the program, we just did a visual confirmation that the resulting paragraph was correct and that the program didn't crash with a variety of common keyboard strokes.

```
Input a Name: Jasque
Input a Vegetable: Carrot
Input a Candy: Bubblegum
Input a Capital Name: New York
Input a Spice: Pepper
Input a Job Title: Peasent
Input an Adjective: Blue
Input a Gourd: Pumpkin
Input a Celestial Body: Saturn
Input an Emotion: Happy
Input a Liquid: Water
Input a Store: WalMart

Once upon a time, Jasque was magically transported
to a magical kingdom on the brink of collapse. The mighty
Carrot have clashed with the Bubblegum Empire and
driven the Empire into their walls of their most mighty city,
New York, which has definitely fended of countless invasions
in years past. But the Carrot were like no past invading
force, bringing with them the power of Pepper, they blasted
at the mighty walls, reducing them to rubble, yet the Bubblegum
people fought back to resist the irresistible tide of the
Carrotwashing over them.

Jasque was taken to the last Peasentof the Bubblegum
people, who pleaded with Jasque to fight alongside him
and give his Empire a fruitless chance to see another
day. This man agreed, and went out to battle to give victory to
theBubblegum people. But after hours of Blue
fighting, Jasque finally saw the commander of the
Carrot forces, an imposing Pumpkin with Pepper
baked into his pores, thundering to his forces that
Saturn has promised them victory this day.

The scent wafting off of the commander sparked
the Happy inside of Jasque, who went
mad and started eating all of the nearby soldiers.
Happy, never ceasing, gnawed at the insides of
Jasque, and the more Jasque ate, the more it
grew. The nearby soldiers panicked at the sight of the
man, attempting to flee in vain as Jasque ate
more and more of their kin. In short order, the battlefield
is silent, as all the competing food now laid in rest,
slowly dissolving all away in the acidic Water
of the stomach of Jasque, who now slept in peace
inside of WalMart.
```

4. Reflection

For the problem we were given and the intended size of this project, we believe we arrived at the best solution. We consider other possible ways we could implement this same program in the Planning section, but those methods had caveats that outweighed the benefits they would bring to the project. That being said, this program is not scalable in anyway, and if this program were to be built upon, then pursuing the other possible ways of implementing this program would be worth pursuing, specifically using a separate class for the input and output functionality of the program, making extensive use of lists and loops, and potentially outputting to a file.