# Exercise 1: MadLibs

Refactor the last exercise. Write a program that reads a MadLib story description from a text file and prompts the user to enter the fill-ins. Then it prints the story filling in the inputs in the text. Use the STL map for linking names to inputs. Use exceptions and report file errors.

A sample story is shown below. A single “%” on a line separates the inputs from the story. You should ignore blank lines in the input section.

AliensInClass.txt

%person% Tell me the name of someone you know.

%animal% Name an animal.

%compLanguage% What is your favorite programming language?

%animal2% Name a small animal.

%

\*\*\*\* ALIENS IN THE CLASSROOM!!! \*\*\*\*\*

\*\*\*\*\*\* a madlib story \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

It started out an uneventful %compLanguage% class that evening.

But then %person% pulled a live %animal% out of a bag and

began to eat it. We tried to stop %person% but it was too late

for the %animal%.

"TAKE ME TO YOUR LEADER," %person% said pulling out a %animal2%

and pointing it like a weapon.

#include <iostream>

#include <fstream>

#include <string>

#include <map>

using namespace std;

int main(int argc, char\*\* argv) {

// The STL class does all the magic

map<string,string> userInputs;

fstream storyFile;

storyFile.exceptions(ifstream::failbit | ifstream::badbit);

try {

storyFile.open(argv[1], fstream::in);

string s;

while(!storyFile.eof()) {

getline(storyFile,s);

if(s.size()==0 || s[0]!='%') continue;

if(s=="%") break;

int secondPercent = s.find('%',1);

string key = s.substr(0,secondPercent+1);

string prompt = s.substr(secondPercent+2);

cout << prompt << " :";

string value;

cin >> value;

userInputs[key] = value;

}

cout << endl << endl << endl;

while(!storyFile.eof()) {

getline(storyFile,s);

while(true) {

int per = s.find('%');

if(per<0) break;

int per2 = s.find('%',per+1);

int keyLength = per2-per;

string first = s.substr(0,per);

string key = s.substr(per,keyLength+1);

string second = s.substr(per2+1);

// TODO check to see if it is found

s = first+userInputs[key]+second;

}

cout << s << endl;

}

storyFile.close();

} catch (ifstream::failure e) {

cout << "Error with file I/O. " << endl;

}

}