Compile and execute a Pascal program

The purpose of this exercise is to give you a crash course in Pascal programming by looking at some code. Use one of the following online Pascal interactive development environments (IDE):

- https://www.jdoodle.com/execute-pascal-online
- http://rextester.com/l/pascal_online_compiler

All files referred to below are found in the course Git repository.

There are 2 .pas files, *EmployeeListing.pas* and *Presidents.pas*
Each when run with their respective inputs must produce the outputs provided.
There are small issues with each program. You are to find and correct those issues.

You may use the use the online IDEs and compilers and cut and paste the code and input. You can also install the Pascal Free Compiler on Linux and compile and run from the command line *(sudo apt install fp-compiler-3.2.2*). VSCode is also an option with the installation and configuration of a couple of extensions.

Some Notes on Pascal text input

If *ch* is a character variable, then *read(ch)* will read the next character from the standard input. If *n* is an integer variable, then *read(n)* will read the next integer value from the standard input. A call to *readln* will skip the rest of the current input line. Then the next call to read will read starting on the next input line.
Boolean functions *eoln* and *eof* without parameters test whether the standard input is at the end of the current line or at the end of file, respectively. (If you are typing input data into a Windows command window, a line containing only control-Z is the end-of-file marker. On Linux and the Mac, it's control-D.)

Some notes on Pascal's built-in string type

Read online how to use the string type:

- https://wiki.freepascal.org/String
- https://www.freepascal.org/docs-html/rtl/sysutils/stringfunctions.html

To use Pascal string functions, you may need to add the below line at the beginning of your program: *uses sysutils;*

Deliverables

The two corrected pascal files: *EmployeeListing.pas, Presidents.pas*

HW Submission Instructions

- You must check in all submitted assignments into the YU GitHub system into your own repo, into a special branch that you will create for this course. In order to start with an empty branch, we are going to branch off of the initial commit in your repository.

- Open a terminal from within a folder in which you have cloned your repository and execute the following Git command, which will return the initial commit.

  ```
  git rev-list --max-parents=0 HEAD
  ```

- Verify that it is the initial branch:
  https://github.com/Yeshiva-University-CS/**repo name**/tree/**commit hash**

- Create a new branch named: **`Compilers`**
  - This can be done either via the Webpage
  - Or by executing the following Git command from the same folder::

    ```
    git checkout -b Compilers commit_hash
    git push -u origin Compilers
    ```

- For the balance of the course, you will check out and work on the *Compilers* branch, as opposed to the *main* branch.

- You will check each homework assignment into its own folder.
- All your files for this homework must be in the **/hw1** directory