CPSC 3400 Languages and Computation
Winter 2018

Homework 3
Due: Monday, February 5 at 10:55am

Create a Python program (`hw3.py`) that simulates a Mark-Sweep garbage collection algorithm. In this program, named pointers are referred to using variable names such as "`p`", "`stackPtr`", "`temp3`", etc. Heap blocks are referred to using integers.

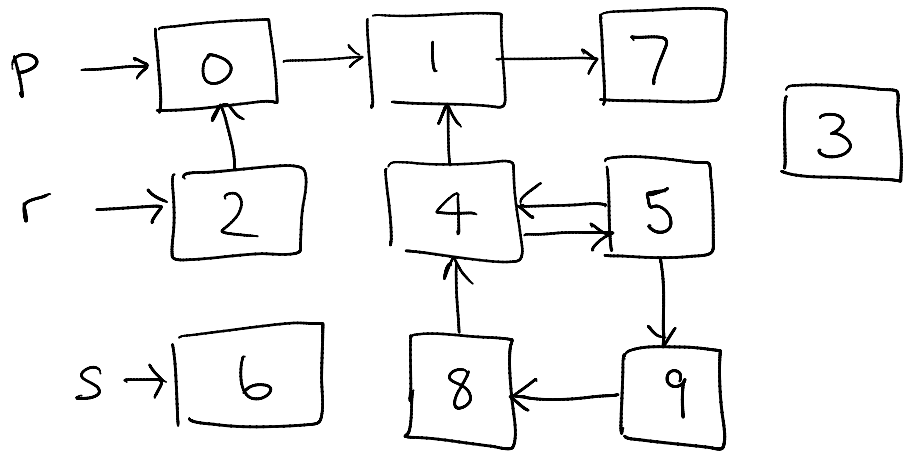
The program has the following functionality:

1. Get the name of an input file from the command line (using `sys.argv`). WARNING: Do not prompt the user for a file name.
2. Process the file. The first line will contain *n* the number of heap blocks – the heap blocks will be identified using the numbers 0 through *n* – 1. Each subsequent line will contain an ordered pair either in the form:

- *named pointer,heap block* (Example: `p,10` p points to heap block 10)
- *heap block,heap block* (Example: `7,3` heap block 7 points to heap block 3)

3. Perform the mark-sweep algorithm.
4. Output which heap blocks are marked and which heap blocks should be reclaimed (swept).

Example Diagram:



Sample Input File:

```
10
p,0
0,1
1,7
r,2
2,0
4,1
4,5
5,4
5,9
s,6
8,4
9,8
```

Output:

```
Marked nodes: 0 1 2 6 7
Swept nodes: 3 4 5 8 9
```

Notes:

- The sample input file is at `/home/fac/elarson/cpsc3400/hw3/sample.txt`
- You may assume the input is valid and properly formatted. Note: there are no spaces in an input file line.
- A valid variable name consists of letters, digits, and underscores but cannot begin with a digit. (Remember that the program will only be tested with valid variable names.)
- The output must print the marked nodes and swept nodes in numerical order separated by spaces.
- Your algorithm must run in polynomial time but does not need to optimal.

Grading

The grading breakdown is as follows for a total of 50 points:

Mark-sweep algorithm	*40 points*
• Correct output for same file (`memory.txt`):	12 points
• Seven additional tests	28 points (4 points each)
Programming style / proper Python usage	*10 points*

For programming style, make sure you are making appropriate use of Python data structures and functions. Also make sure you are writing readable and organized code. See the programming assignment expectation handout.

Additional grading notes:

- If the output is correct but hard-to-read, you may lose points.
- Programs that contain syntax errors will receive a zero.

Submitting your Assignment

On `cs1`, run the following script in the directory with your program:

```
/home/fac/elarson/submit/cpsc3400/hw3_submit
```

This will copy the files `hw3.py` to a directory that can be accessed by the instructor. Please be sure to keep the same file names or the submission program will not work. Only the last assignment submitted before the due date and time will be graded. ***Late submissions are not accepted and result in a zero.***