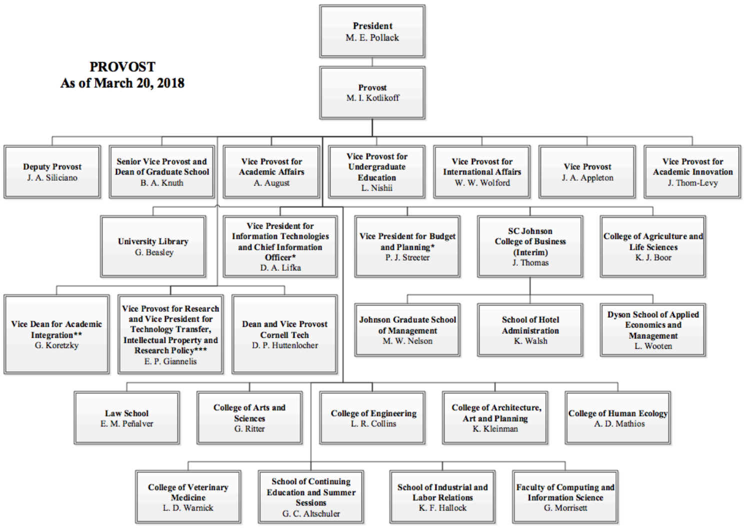
CS 1110 Spring 2018, Assignment 4, phase 2: Org charts*

Version 2, April 12: added appendix

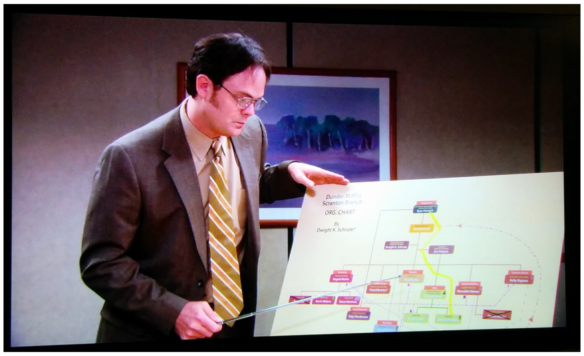
1 Motivation, or, "You're not the boss of me"

The "org chart": if you're part of a large institution, the day may come when you suddenly become very interested in some aspect of these representations of supervisor and subordinate relationships.[1] Cornell's own collection of org charts[2], quoting M.G. Jacobides (2007), waxes poetic on the subject, saying that these objects can be "considered as the viewing glass or perspective through which individuals see their organization and its environment". If so, what does the Provost's chart[3] say about Cornell?


PROVOST
As of March 20, 2018
President
M. E. Pollack
Provost
M. I. Kotlikoff
Deputy Provost
J. A. Siliciano
Senior Vice Provost and Dean of Graduate School
B. A. Knuth
Vice Provost for Academic Affairs
A. August
Vice Provost for Undergraduate Education
L. Nishii
Vice Provost for International Affairs
W. W. Wolford
Vice Provost
J. A. Appleton
Vice Provost for Academic Innovation
J. Thom-Levy
University Library
G. Beasley
Vice President for Information Technologies and Chief Information Officer*
D. A. Lifka
Vice President for Budget and Planning*
P. J. Streeter
SC Johnson College of Business (Interim)
J. Thomas
College of Agriculture and Life Sciences
K. J. Boor
Vice Dean for Academic Integration**
G. Koretzky
Vice Provost for Research and Vice President for Technology Transfer, Intellectual Property and Research Policy***
E. P. Giannelis
Dean and Vice Provost Cornell Tech
D. P. Huttenlocher
Johnson Graduate School of Management
M. W. Nelson
School of Hotel Administration
K. Walsh
Dyson School of Applied Economics and Management
L. Wooten
Law School
E. M. Peñalver
College of Arts and Sciences
G. Ritter
College of Engineering
L. R. Collins
College of Architecture, Art and Planning
K. Kleinman
College of Human Ecology
A. D. Mathios
College of Veterinary Medicine
L. D. Warnick
School of Continuing Education and Summer Sessions
G. C. Altschuler
School of Industrial and Labor Relations
K. F. Hallock
Faculty of Computing and Information Science
G. Morrisett

* Dual report to Joanne DeStefano, Executive Vice President and Chief Financial Officer
** Dual report with A. M. K. Choi, Provost for Medical Affairs and Dean of Weill Cornell Medical College
***Reports to President in the capacity of Vice President for Technology Transfer, Intellectual Property and Research Policy

What does Dwight K. Schrute's chart[4] tell us about Dunder Mifflin and about him?



[*] Authors: Lillian Lee.

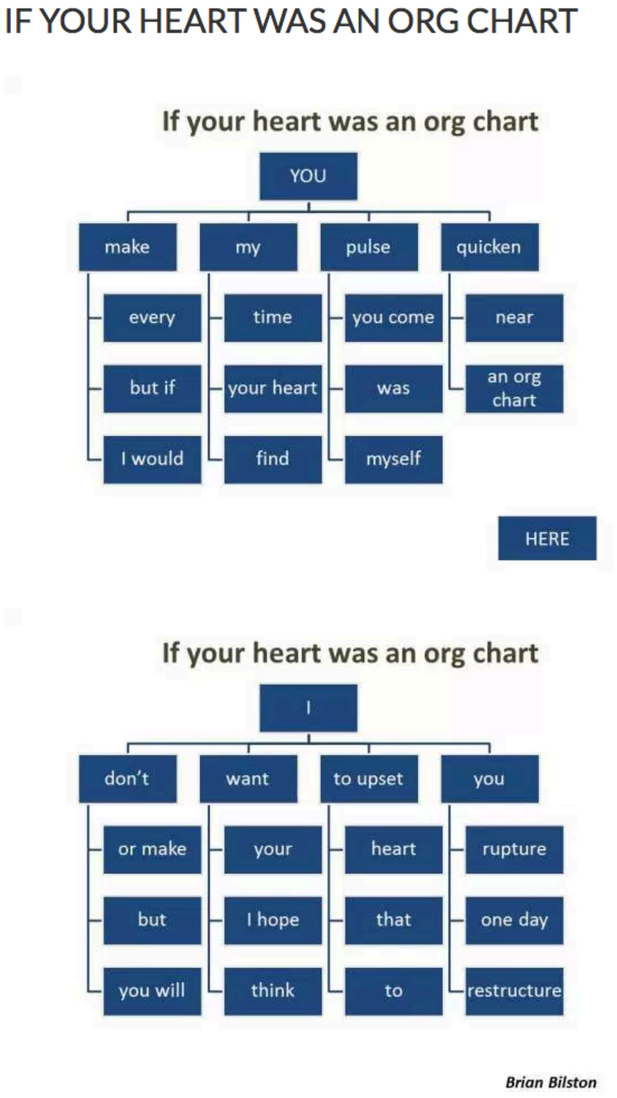
[1] Real questions: "How many people are depending on me?" "What if I need to go over someone's head on this matter?" "Where does the buck actually stop?"

[2] http://dbp.cornell.edu/university-org-structure/

[3] https://cpb-us-e1.wpmucdn.com/blogs.cornell.edu/dist/b/6138/files/2015/09/1000006-Provost-03202018-2fuudon.pdf

[4] https://imgur.com/gallery/Y8EvG

And what does it say about the ubiquity of the org-chart form that the following poem[5] exists?



Contents

[5] https://brianbilston.com/2015/11/22/if-your-heart-was-an-org-chart/

2 Instructions

We assume you have completed Phase 1. Download the A4 Phase 2 files[6] from the course Assignments page. Complete the functions in a4.py following the directions in that file.

2.1 Advice

We've provided a4_lightchecks.py to help you along in the testing process. See the instructions in that file.

That file gets test inputs by utilizing Positions that we've created for you in files example_chart_cornell.py and example_chart_scraggly.py To get a visualization of the relationships created by either file, run Python on it. Section 5 gives an example. You can use the code in those files as inspiration to construct other test sets of inter-related Positions.

For debugging, you may find it useful to print relevant Positions. You may also visualize all the Positions "beneath" a given Position p by the method call p.draw(). Additionally, in debugging recursive functions, it is often useful to check the types of the inputs.

As mentioned in Phase 1, we wrote a recursive function for you in file positions.py that you may wish to use as a model.

We also hope you find time to learn some useful techniques by reading other parts of the code we provided.

3 Rules (all the same as for previous assignment)

There is no revise-and-resubmit for this or any subsequent assignment unless otherwise noted.

3.1 You need to re-group (so to speak) on CMS, regardless of prior groupings

You may work alone or with just one other person, who can be someone you've grouped with before in CS1110, or a different person.

If you are partnering, regardless of whether you were grouped in a previous assignment, the two of you must still form a new group *specifically for this assignment* on CMS before submitting.[7]

If your partnership dissolves, see the course Academic Integrity description about "group divorce" for what to do.

3.2 (Dis-)allowed collaborations and documentation requirements

Our policies are laid out in full on the course Academic Integrity page, but we re-state here **the main rules**: where "you" means you and, if there is one, your one CMS-registered group partner,

1. Never look at, access or possess any portion of another group's work in any form.
2. Never show or share any portion of your work in any form to anyone except a member of the course staff.
3. Never request solutions from outside sources; for example, on online services like StackOverflow.
4. DO specifically acknowledge by name all help you received, whether or not it was "legal" according to (1)-(3).

4 Code cleanup and submission checklist (all the same as before except the due date)

Before submitting, ensure your code obeys the following.[8]

- Lines are short enough (˜80 characters) that horizontal scrolling is not necessary. (We've sometimes violated this for readability.)
- You have indented with spaces, not tabs (this is not an issue if using Komodo Edit).

[6] There have been very slight changes to the Phase 1 versions.
[7] This links your submission "portals".
[8] One reason for these requirements is that they speed up the process of reading hundreds of files.

- You have removed any debugging `print` statements.
- You have removed all `pass` statements that were provided with the initial code.
- You have removed "instruction" comments, typically marked "`# STUDENTS:`".
- If you added any helper functions, these have good docstring specifications and you have put sufficient testing code for your functions.

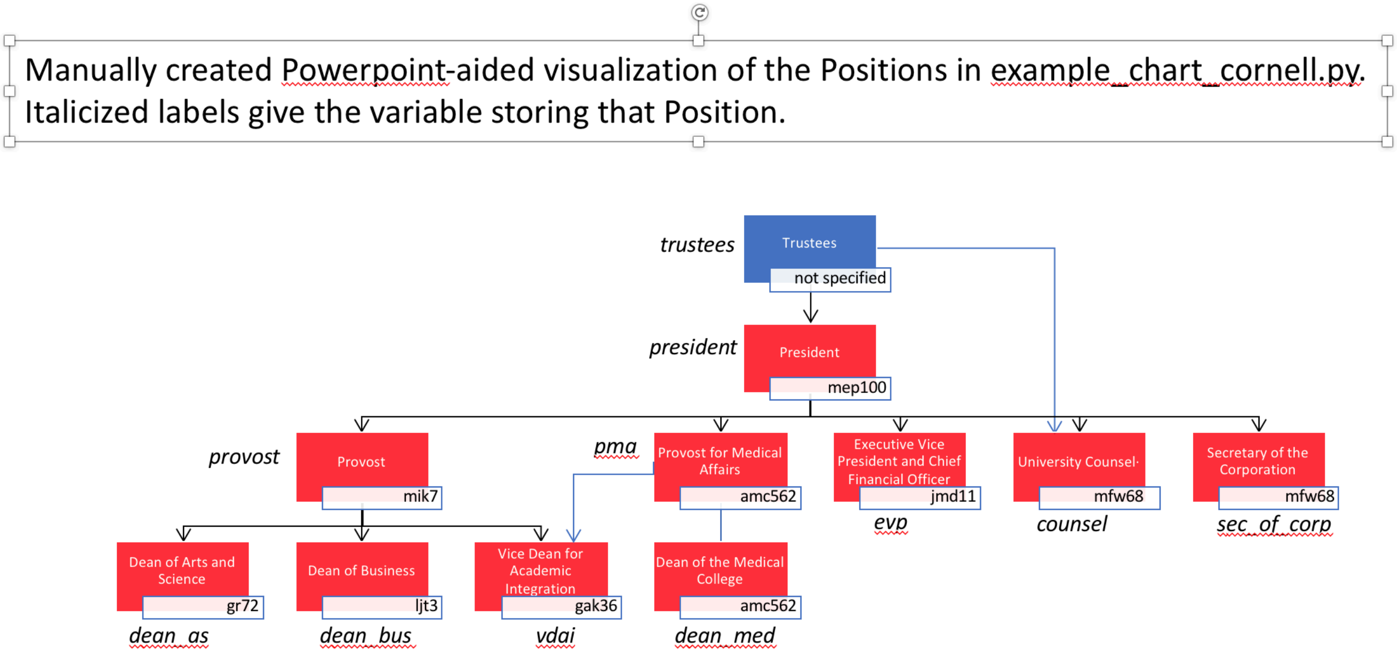
Make sure the following are all true before you submit.

1. You (and your partner) have included your name(s) and NetID(s) in the header of all files.
2. You've also indicated in the file headers the names of any (non-staff) people whose help should be acknowledged.
3. The date in the header comments has been changed to when the files were last edited.
4. You have set your CMS notifications settings to receive email regarding grade changes, and regarding group invitations.

4.1 Due dates

1. If you are partnering: well before submission, follow the instructions in the "How to form a group" instructions on the course Assignments page. Both parties need to act on CMS in order for the grouping to take effect.
2. By 2pm on Thu Apr 19, submit whatever you have done at that point to CMS, following steps 1-3 in the "Updating, verifying, and documenting assignment submission" section of the course Assignments page. It is OK if you haven't finished working on the files yet.
3. By **11:59pm on Thu Apr 19**, make your final submission, again following the aforementioned steps 1-3.[9]

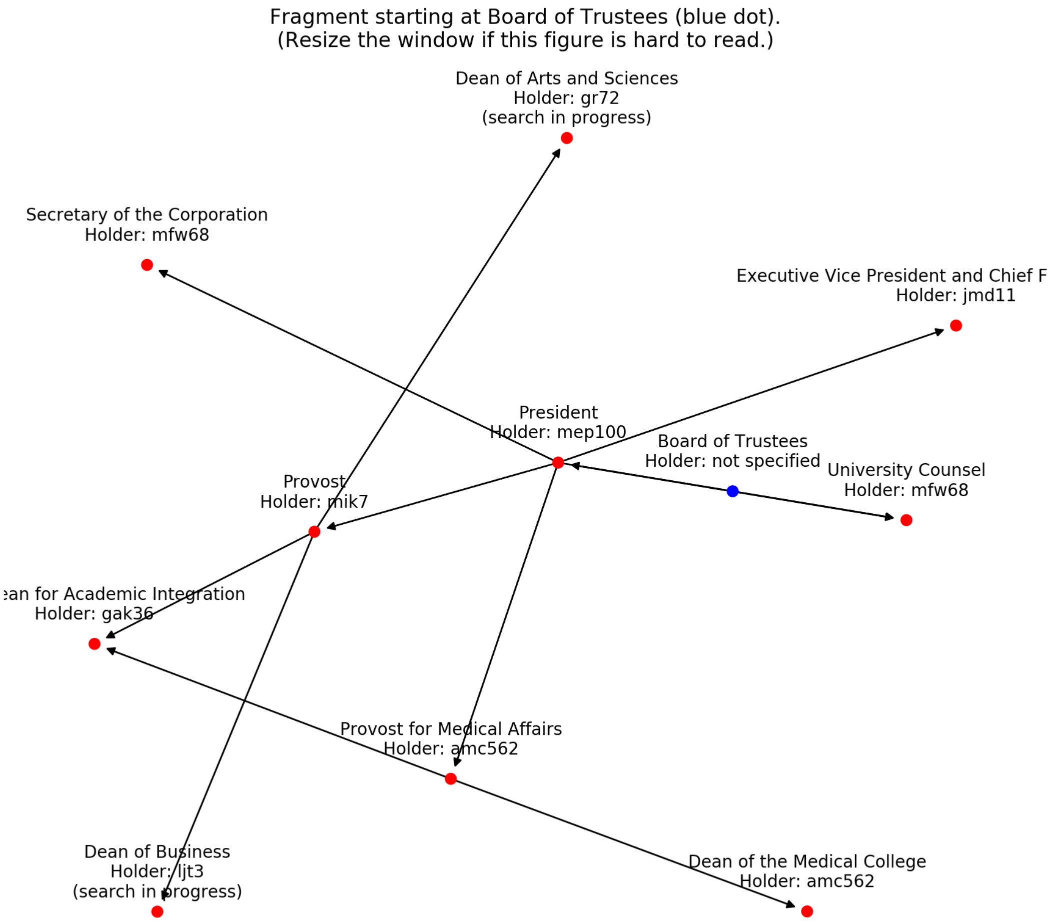
5 Appendix: visualizations of the example Positions created in example_chart_cornell.py



Manually created Powerpoint-aided visualization of the Positions in example_chart_cornell.py. Italicized labels give the variable storing that Position.

[9] The 2pm checkpoint on Thu Apr 19 provides you a chance to alert us during business hours if any problems arise. Since you've been warned to submit early, do not expect that we will accept work that doesn't make it onto CMS on time, for whatever reason. There are no so-called "slipdays" and there is no "you get to submit late at the price of a late penalty" policy. Of course, if some special circumstances arise, contact the instructor(s) immediately.

Alternately, you can visualize the same positions automatically via a call like you see in example_chart_cornell. In that file, trustees is a Position, and here a possible result of positions.draw(trustees):


Fragment starting at Board of Trustees (blue dot).
(Resize the window if this figure is hard to read.)
Dean of Arts and Sciences
Holder: gr72
(search in progress)
Secretary of the Corporation
Holder: mfw68
Executive Vice President and Chief F
Holder: jmd11
President
Holder: mep100
Board of Trustees
Holder: not specified
University Counsel
Holder: mfw68
Provost
Holder: mik7
ean for Academic Integration
Holder: gak36
Provost for Medical Affairs
Holder: amc562
Dean of Business
Holder: ljt3
(search in progress)
Dean of the Medical College
Holder: amc562

As you can see, the drawing package being utilized is not optimized for lovely layout, Also, it is often hard to see that for the Cornell example, the University Counsel reports to both the Board of Trustees and the President. You might need to re-run Python on a file several times to get a satisfactory image; the layout will change each time.

But, to visualize the set of Positions directly or indirectly subordinate to a new Position p you create, you can run positions.draw(p) if you've imported module positions.