

Assignment 1: Algorithms as Decision Makers

This notebook provides starter code and a template for formatting your work.

Tasks 1-3 and 5 require textual answers. Markdown cells have been provided for these. I have left them in an editable state, but at any time you can format them by clicking on them and pressing `shift+enter`. Double clicking on the text will return it to the editable state. Running all cells will also format everything and run your code. This is what I'll be doing duringn grading so it really doesn't matter what state cells are in when you hand in your assignment.

Note that in Markdown starting a line with an `` will create a bullet and indenting will create a sub bullet. A full list of formatting commands is available here: <https://www.datacamp.com/community/tutorials/markdown-in-jupyter-notebook>*

Please rename this file, replacing `YourName` with your first and last name, and also write your name in the cell below:

Kerry Liu

Task 1: Assess User Needs

10 Factors that matter

1. Housing Type (Depending on Marriage or Parenthood)
 - My research shows that many universities offer different types of housing for single students, students with a spouse, and students with children. Thus, I would like to offer these three types of housing in my program based on whether having a spouse or children to live with.
 - The assignment of different types of housing should be decided separately. Only students who are applying for the same type of housing should be ranked.
 - The output of this program should be a combination of Housing Type and Score (e.g., Single Students Housing, 15 points)
2. Enrollment Status (Matriculated / Non-matriculated)
 - Priority should be given to students who are enrolled in a degree program and presently seeking a degree.
 - I will refer them as "matriculated students" in the rest of this assignment.
3. Class Year
 - I assume that the concept of "Class Year" is only applicable to matriculated students.
 - Freshmen should be prioritized for being in an unfamiliar environment.
 - Other students who are closer to graduation should have preference.
 - Not calculated by credit completed, because the number of credits required differs from program to program.
 - I assume that all programs are 4 years in length, and I will use a simple equation here to calculate the bonus point according to class year.
 - If a number larger than 4 is entered, than no points will be added.
4. Full time / Part time
 - Among matriculated students, those enrolled full-time should be given more priority.
5. Academic Suspension

- If a student is on Academic Suspension, he or she is considered non-matriculated.
 - Among all non-matriculated students, those who are on academic suspension due to a failure to maintain good academic standing should receive fewer points.
6. Student source
- Out-of-province students and international students should be given priority.
7. Age
- Senior citizens (60 or older) should have preference and receive special care.
8. Disability (Physical or Mental)
- Students with disabilities should be able to choose housing that accommodates those issues.
9. Academic Integrity
- Those who have never violated academic integrity should be given priority over those who have.
10. Conviction
- Those who have never been convicted of a crime should be given priority over those who have.

Task 2: Design Your Algorithm

- **Q1** *Do you have any legally dependent children living with you? (y/n)*
 - **Yes** : Housing Type = Students with Children Housing
 - **No** :
 - Q1.1** *Do you have a spouse living with you? (y/n)*
 - **Yes** : Housing Type = Couples without Children Housing
 - **No** : Housing Type = Single Student Housing
-
- **Q2** *Are you a matriculated student? (y/n)*
 - **Yes** :
 - Q2.1** *What year are you? (1/2/3/4/...) [This question involves equation]*
 - **1** : +4 points
 - **2** : +1 points
 - **3** : +2 points
 - **4** : +3 points
 - **>4** : pass
 - Q2.2** *Do you study full-time? (y/n)*
 - **Yes** : +4 points
 - **No** : +2 points
 - **No** :
 - Q2.3** *Are you on Academic Suspension? (y/n)*
 - **Yes** : -4 points
 - **No** : pass
-

- **Q3** *Are you an out-of-province student or international student? (y/n)*

- **Yes** : +1 points
 - **No** : pass
-

- **Q4** *Are you a senior citizen? (y/n)*
 - **Yes** : +5 points, *Special care* = True
 - **No** : pass
 - **Q5** *Do you identify as a person with a visible or non-visible disability? (y/n)*
 - **Yes** : +5 points, *Special care* = True
 - **No** : pass
-

- **Q6** *Have you ever violated academic integrity? (y/n)*
 - **Yes** : -4 points
 - **No** : pass
 - **Q7** *Have you ever been convicted of a crime? (y/n)*
 - **Yes** : -4 points
 - **No** : pass
-

Task 3: Test Your Algorithm

Summary of findings

Replace with your answer for Task 3 (summary of findings)

I tested my algorithm on my roommate, my brother, and my cat. Testing on my cat made me realize that I didn't account for the unique needs of feline graduate students.

Test Cases

No.	Spouse	Child	Enrollment Status	Year	Source	Age	Disability	Other Issues	Housing Type Output	Score Output
1	✓	✓	Full-time Matriculated	1	Out-of-province	19	X	None	Students with Children Housing	9
2	✓	X	Part-time Matriculated	3	Local	21	✓	Academic Dishonesty	Couples without Children	5
3	X	✓	Non-Matriculated	N/A	Local	20	X	None	Students with Children	0
4	X	X	Full-time Matriculated	4	International	22	X	None	Single Students Housing	8
5	X	X	Part-time Matriculated	1	Local	62	X	None	Single Students Housing	11
6	X	✓	Full-time Matriculated	2	Local	20	X	Conviction	Students with Children Housing	1
7	✓	X	Full-time Matriculated	4	Local	19	✓	None	Couples without Children	12
8	X	X	Academic Suspension	N/A	Local	25	X	None	Single Students Housing	-4

Task 4: Write Code to Automate Your Algorithm

Remember to first run the starter code and make sure you understand what's happening. Then modify the code to make it follow your algorithm

```
# Modify with your answer for Task 4

# Created by: Evan Peck (Bucknell University)
# Modified by: Karyn Moffatt (McGill University)
# Modified by: Kerry Liu (McGill University)
# - Contact: karyn.moffatt@mcgill.ca
# - Contact: xin.liu6@mail.mcgill.ca
# - Last Modified: Feb 20, 2021

# Keeps track of the point total during questions
score = 0

# A header to start the program
print("-----")
print("      HOUSING SCORE CALCULATOR      ")
print("-----")
print()

# I am using while loop with "break" and "continue" for every factor.
# so that users don't have to start over if there's a typo in the middle.
# I am trying to be "user-centered", although it makes my program harder to read.

# Deciding Housing Type Depending on Marriage or Parenthood
```

```

while True:
    q1 = input(
        'Do you have any legally-dependent children living with you? (y/n) \n>
    ')
    # Students with children regardless of marriage status
    # should apply for "Students with Children Housing".
    if q1.lower() == 'y':
        housing_type = 'Students with Children'

    # For students without children:
    elif q1.lower() == 'n':
        q1_1 = input('Do you have a spouse living with you? (y/n) \n> ')
        # married students will be considered "Couples without Children
Housing".
        if q1_1.lower() == 'y':
            housing_type = 'Couples without Children'
        # single students will be considered "Single Students Housing",
        elif q1_1.lower() == 'n':
            housing_type = 'Single Student Housing'
        else:
            print("Please enter 'y' or 'n'.")
            continue
    else:
        print("Please enter 'y' or 'n'.")
        continue
    break

# Enrollment Status (Matriculated / Non-matriculated)
while True:
    q2 = input('Are you a matriculated student? (y/n) \n> ')
    if q2.lower() == 'y':
        # class year (only applies for Matriculated)
        q2_1 = input('What year are you? (1/2/3/4/...) \n> ')
        # used "try" to test whether the input is a number
        try:
            q2_1 = int(q2_1)
        except:
            print("Please enter a number.")
            continue
        else:
            if q2_1 == 1:
                score += 4
            elif q2_1 in [2, 3, 4]:
                score += q2_1 - 1
        # full time / part time (only applies for Matriculated)
        q2_2 = input('Do you study full-time? (y/n) \n> ')
        if q2_2.lower() == 'y':
            score += 4
        elif q2_2.lower() == 'n':
            score += 2
        else:
            print("Please enter 'y' or 'n'.")
            continue

    elif q2.lower() == 'n':
        # Academic Suspension (only applies for Non-matriculated)
        q2_3 = input('Are you on Academic Suspension? (y/n) \n> ')
        if q2_3.lower() == 'y':

```

```

        score -= 4
    elif q2_3.lower() == 'n':
        pass
    else:
        print("Please enter 'y' or 'n'.")
        continue
else:
    print("Please enter 'y' or 'n'.")
    continue
break

# Student Source
while True:
    q3 = input(
        'Are you an out-of-province student or international student? (y/n) \n>
    ')
    if q3.lower() == 'y':
        score += 1
    elif q3.lower() == 'n':
        pass
    else:
        print("Please answer 'y' or 'n'.")
        continue
    break

# for people who might need special care
# a note will appear at the end
special_care = False

# Age
while True:
    q4 = input('Are you a senior citizen? (y/n) \n> ')
    if q4.lower() == 'y':
        score += 5
        special_care = True
    elif q4.lower() == 'n':
        pass
    else:
        print("Please enter 'y' or 'n'.")
        continue
    break

# Disabilities
while True:
    q5 = input(
        'Do you identify as a person with a visible or non-visible disability?
    (y/n) \n> ')
    if q5.lower() == 'y':
        score += 5
        special_care = True
    elif q5.lower() == 'n':
        pass
    else:
        print("Please enter 'y' or 'n'.")
        continue
    break

```

```

# Academic Integrity
while True:
    q6 = input('Have you ever violated Academic Integrity? (y/n) \n> ')
    if q6.lower() == 'y':
        score -= 4
    elif q6.lower() == 'n':
        pass
    else:
        print("Please answer 'y' or 'n'.")
        continue
    break

# Conviction
while True:
    q7 = input('Have you ever been convicted of a crime? (y/n) \n> ')
    if q7.lower() == 'y':
        score -= 4
    elif q7.lower() == 'n':
        pass
    else:
        print("Please enter 'y' or 'n'.")
        continue
    break

# At the end of the program, tell the user their housing type and score
print()
print("-----RESULT-----")
print('Housing Type:', housing_type)
print('Housing Points Score:', score)
print("-----")

# a note for senior citizens and those with disabilities
if special_care:
    print('*Please contact us as soon as possible so we can best meet your needs.*')

```

```

-----
HOUSING SCORE CALCULATOR
-----

```

```

Do you have any legally-dependent children living with you? (y/n)
> n
Do you have a spouse living with you? (y/n)
> y
Are you a matriculated student? (y/n)
> n
Are you on Academic Suspension? (y/n)
> y
Are you an out-of-province student or international student? (y/n)
> n
Are you a senior citizen? (y/n)
> y

```

```
Do you identify as a person with a visible or non-visible disability? (y/n)
> n
Have you ever violated Academic Integrity? (y/n)
> y
Have you ever been convicted of a crime? (y/n)
> n
```

```
-----RESULT-----
Housing Type: Couples without Children
Housing Points Score: -3
-----
*Please contact us as soon as possible so we can best meet your needs.
```

Task 5: Test Your Assumptions

Replace with your answer for Task 5 (two paragraph reflection on your algorithm's strengths and weaknesses)

When considering whether to give priority to married students and students with children, I did some research and realized that many universities offer different types of housing according to marriage and parenthood. Since a single student is not eligible for "Students with Children Housing" in the first place, there is no point to rank them together. Students eligible for different types of housing should be ranked separately. I will first ask if a student has children to live with. Students with children regardless of marriage status should apply for "Students with Children Housing". For students without children, single students will be considered "Single Students Housing" while those with a spouse should apply for "Couples without Children Housing". If the proportion between different types of housing is similar to the proportion between different types of students, then it can be considered fair.

Besides, to simplify the problem, I assumed that the length of each program is 4 years. In many universities, undergraduate housing is sequentially allocated to students in their senior, junior, sophomore, and freshman year. For other universities like McGill and UBC, incoming freshmen are guaranteed student housing. Every student will go through year 1 to year 4, and somewhere they will be prioritized. In this sense, any practice is fair, because they will receive the same points all together over the four years. Still, I choose to keep a balance between these two practices, by giving the most priority to freshmen, and then those closer to graduation. There are circumstances when a student is on year 5, maybe due to reasons like academic suspension. In my program, input larger than 4 gets no score, because the principle remains that every student gets the same total points over their undergraduate studies, which is the fairest to me.

I recognized three weaknesses for my program. First, there are often more than three types of student housing in real life scenarios. For example, single students can choose from studio, 2-bedroom apartment, town house, etc. The housing types differ from university to university. Second, some people might not feel comfortable to answer some of the questions like those concerning age and disabilities issues. Due to privacy reasons, an option of "not willing to provide answers" should be given. However, there is a conflict that if we do not know their difficulties, we cannot help them and given them extra points. Third, I imagine the major of students would have the same answer for all the questions except class year and student source. That means that most students' score will fall into the range of 5 to 9. If the threshold falls into the same range, then this program is not enough to decide. Assume that there is 15% percent of students for each

score from 5 to 9, which is a lot of students, then probably, we need to further draw lots to decide who can live in student housing.

Don't forget to rename the file with your name. Upload your finished assignment to MyCourses - Assignment 1