# **Project 1:** Developers as Decision Makers

*A modified version [of the assignment created](https://ethicalcs.github.io/" \l "decision-makers) by Evan Peck at Bucknell University*



*In this lab, we will create algorithms that determine housing for SFSU students.*

When we say the word ‘algorithm’, we tend to ascribe agency to the computer. It is *deciding* things for us. But the reality is that there is no magic. There are software developers like you and me who design and create sets of rules that the computer carries out for us.

**These algorithms are all around us, and they are *constantly* making decisions.**

The decisions we make in code impact the lives of real people. For example, the [**Silicon Valley Triage Tool**](https://economicrt.org/publication/silicon-valley-triage-tool/) is an *algorithm* that identifies homeless people for whom giving them housing would cost the public less than keeping them homeless. So even as we learn the simple structures of code, we need to think about *how we can make* ***good*** *decisions.*

We are going to explore this idea in a more familiar context to you - **university housing allocation**. At SFSU, on-campus housing is in high demand but very limited. This has resulted in the university relying on a first-come first-serve policy: <https://housing.sfsu.edu/apply>. You might not think of it as one, but this method is **an algorithm.**

In this project, you will have the opportunity to design your *own* algorithm. We’re also going to begin dabbling with a [human-centered design process](http://www.designkit.org/human-centered-design) to make sure that **the decisions we make are never untethered from the people we impact.**

### In this project, you’ll practice...

* translating English rulesets into code,
* soliciting text input from people,
* **\*Applying conditionals (if, else if, else) to make decisions with a program\*,**
* using an accumulation variable to keep track of information in a program, and
* integrating basic human-centered design processes into your programs.

## First: Is the first-come first-serve fair? Can we do better?

[R1] You must start by thinking if the existing way of things doing is fair. Is the first-come first-serve approach to housing allocation fair? Notice how you must think about what “fairness” involves.

I feel that the first-come first-serve approach is fair for student housing. If you do not apply for housing as soon as it is available, it’s safe to assume that you are not as desperate as other students when it comes to finding housing on campus.

[R2] Who are the actors that may be at a greater disadvantage with the current system?

Other actors that may be at a greater disadvantage might be those who often have a busy schedule have events that interfere with getting housing when it is first available. The current system favors those with a lot more free time.

[R3] Do you think there is a better way to allocate housing such that it is fairer? What ethical framework(s) guide your evaluation?

Even though I feel that first-come first-serve is sufficient, there are other fairer ways to allocate housing. For example, we could take the applicant’s economic background, income, and cultural backgrounds into account.

## The Decision-Maker: Who gets to choose their housing first?

**Your job is to build an algorithm that helps determine the order in which students will get to select their housing.** To simplify things, we’re going to use a *point system.*

* Students are awarded several points based on a variety of factors.
* Students with the most points get first choice at housing.

**This real approach** is used by many universities. For example, consider the following *real* *point system* used by another college in the United States:

* *Current Freshman:* 1 point
* *Current Sophomore:* 2 points
* *Current Junior:* 3 points
* *Current Senior:*4 points
* *23+ Years of Age:* 1 point
* *Full-Time, Off-Campus Program Credit (e.g., student teaching):* 1 point
* *Academic Probation:* -1 point
* *Possible Academic Suspension:* -2 points
* *On Disciplinary Probation at Any Point during the Academic Year:* -3 points

So, a junior (+3 points) who is 23 years old (+1 point) would have priority over a senior (+4 points) who is on academic probation (-1 point).

**Overall goal:** Create a program that assigns points to students to prioritize them in housing selection.

## [R4] Before you Code, Assess the Needs of Your Users

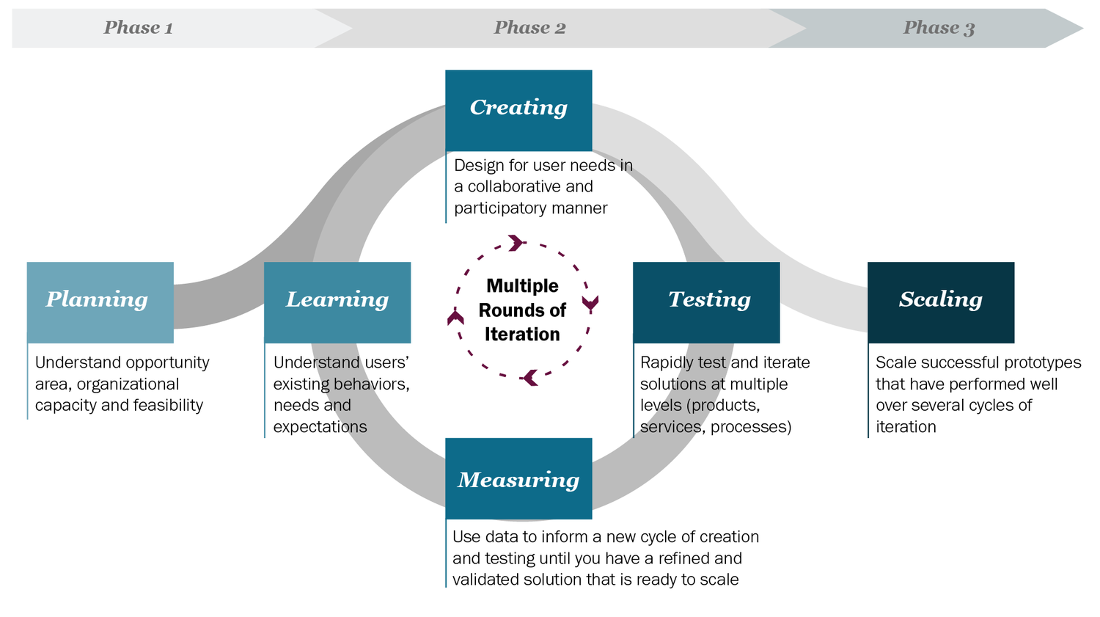
While the list above was *one* college’s take, there are **many** more potential aspects to consider if you want to create a **fair** algorithm that considers the diverse needs of students.

**You should not create a program that serves people without talking to people.** Talk to other students in your class. Ask them about their needs. What other unique factors may be important in deciding who should choose housing first?

**Group Work:** Form a group of 3—4 students. Discuss the various factors to consider when deciding on housing allocation. Fill in the bullet-point list below with the factors that came up in the conversation. Give each one a numeric priority (-1 to 5 points).

* Factor 1: Distance from home 2
* Factor 2: Economic background 4
* Factor 3: Student type (Freshmen, Sophomore, Junior, Senior, transfer, international transfer, etc…) 1
* Factor 4: Disability 3
* Factor 5: Academic standing 5
* Factor 6: Time of application -1
* Factor 7: Familial status 4
* Factor 8: Lottery (If housing is full) 1

Notice that this is the learning phase of the **human-centered design process** shown in the following figure.



Talking to users, prototyping ideas, and then testing them with users, is part of what is known as the **human-centered design process**… a key process for developing useful and usable programs.

## Assessing the Needs of Our Users

## So far, you have discussed the factors amongst yourself. It is not enough; other people may think differently about what is fair.

## Using the list that was prepared in your group, interview other people. Try to interview people outside the CSC 101 class. These can be family, friends, or classmates from a different class. Ask what factors are important to them. Show the factors that you had identified from group work and ask them about their priority.

### **Individual Work:**

## [R5] Results of the Interviews and Discussions: Factors that Matter.

I interviewed 3 SFSU students and 1 family member about their factors (full score can be earned when interviewing 3 or more different people).

**MODIFY THE LIST BELOW WITH OBSERVATIONS FROM YOUR INTERVIEWS.** You can add or delete factors. Provide a number with the factors to indicate which factors that you think are the most important. For example, if you think class year is the most important, place a “1” by that factor. If you think two or more factors are equally important, place the same number by those factors.You can use our factors from the group work to start with.

* Class year
  + Students closer to graduation should have preference.
* Age
  + Older students should have preference.
* Student 1:
  + Class year: Senior
  + Age: 22
  + Ranking:
    1. Distance from home
    2. Economic background
    3. Familial Status
    4. Disability
    5. Academic Standing
    6. Student type
    7. Time of application
    8. Lottery
* Student 2:
  + Class year: Senior
  + Age: 22
  + Ranking:
    1. Economic Background
    2. Academic Status
    3. Familial Status
    4. Disability
    5. Distance from Home
    6. Time of Application
    7. Student Type
    8. Lottery
* Student 3: Junior
  + Class year: Junior
  + Age: 21
  + Ranking:
    1. Disability
    2. Economic Background
    3. Familial Status
    4. Academic Status
    5. Distance from Home
    6. Lottery
    7. Student Type
    8. Time of Application
* Family member (my sister):
  + Class year: Junior
  + Age: 20
  + Ranking:
    1. Distance from Home
    2. Disability
    3. Economic background
    4. Academic status
    5. Familial Status
    6. Student Type
    7. Time of Application
    8. Lottery
* Conclusion:
  + My ranking should look like this:
    1. Economic Background
    2. Distance from Home
    3. Familial Status
    4. Academic Status
    5. Disability
    6. Student Type
    7. Time of application
    8. Lottery

## [R6] Design and Plan Your Point-Assigning Algorithm

Now it is time to translate our student needs into a concrete algorithm. Be careful and limit yourself to the most important factors. *You have a limited amount of time in this class!*

Our overall algorithm will be the following:

1. Ask students questions (like *What class year are you?*).
2. Assign points based on their answers (like *4 points for seniors*).
3. Accumulate their total points across all answers (like *You have 23 housing points*).

**Your goal:** Create a bullet-point list that describes the factors you are considering, and how you are mapping those factors to point values (positive or negative). Please do your best to correctly fill out the list. When coding starts, this will serve as an outline for writing your prompts and how you will handle responses.

## Writing Our Algorithm in English

**(EDIT OR REPLACE THE FOLLOWING LIST TO REFLECT *YOUR* HOUSING PRIORITIES.)**

For our algorithm, we are going to prioritize class year and age.

* Question 1: What year are you (1, 2, 3, 4):
  + 1: +1 pts
  + 2: +2 pts
  + 3: +3 pts
  + 4: +4 pts
* Question 2: How old are you:
  + if greater than 22, then + 1pt
  + if 22 or less, then no points

For my algorithm, we are going to prioritize a multitude of factors

* Are you disabled (Y/N)?
  + Yes: + 2 pt
* How far is your current residence from the school?
  + <= 50 miles: +1 pt
  + <= 100 miles: +2 pts
  + <= 150 miles + 3 pts
  + > 150 miles: + 4 pts
  + I’m from out of state: +3 pts
  + I’m an international student: +2 pts
* What is your current household income level? (Doesn’t take into account how many family members for simplicity).
  + < $50,000 ($30,000 for single, 60,000 for family of 4. I picked a value in between. This is low-income in California): +4 pts
  + < $75,000 (in between low and medium): +3 pts
  + < $100,000 (medium income): +2 pts
  + < $150,000 (medium to high): +1 pts
  + > $150,000 (high): 0 pts
* What is your current academic standing?
  + Dean’s List: +3 pts
  + Good academic standing: +1 pts
  + Academic probation: -1 pts
  + Academic Disqualification: -3 pts
* Are you a full-time student or a part-time student?
  + Full-time: +2 pts
  + Part-time: +1 pts
* What is your seniority in college?
  + Freshman: +1 pts
  + Sophomore: +2 pts
  + Junior: +3 pts
  + Senior: +4 pts
  + I’m a graduate/master’s/PhD student: +5pts

## [R7] Before you Code, Make Sure it Works: User Testing

**How will you know if your program serves people well?** At a minimum, you should have tests considering different groups of people who may be affected by your program.

* Were the results what you expect?
* Did you discover any cases which you haven’t accounted for previously?

**Your goal:** Write **at least** 5 hypothetical *test* cases for your program. You may need to modify the given tests if they do not mention any of your factors. Five test cases must be listed in order to earn a full score for R7.

### Test Cases

**EDIT THE FOLLOWING LIST TO REFLECT *YOUR* PROPOSED POINT VALUES.** You are required to add at least 2 test cases of your own.Make sure to modify the points and/or factors in the already given tests so that they match your factors.

* A full-time disabled student who lives 40 miles away from home. His family has an income of $140,000 a year. He is a sophomore and is good academic standing (9 pts)
* A full-time student who lives 150 miles away from home. Her family has an income of $100,000 a year. She is a freshman and is at good academic standing (9 pts)
* A part-time student who lives 70 miles away from home. His family has an income of $60,000. He is a senior, but is on academic probation (9 pts)
* A full-time student who lives 200 miles away from home. Her family has an income of $80,000 a year. She is a junior and is on the Dean’s List. (14 pts)
* A part time student who lives 45 miles away from home. His family has an income of $75,000 a year. He is a sophomore, but is at risk of academic disqualification. (3 pts)

## [R8] Write Code that Automates Your Decision-Making Process

**Now it’s time to translate your algorithm into code.**

**Your goal:** Implement *the algorithm you designed* in IntelliJ.

* **Step 1:** Follow the instructions posted on Canvas to create a new project for this assignment.
* **Step 2:** Implement your algorithm inside the main method.
* **Step 3:** Is it correct? You should check your code with the test cases you outlined above. You should run your program using the test cases as an outline for inputs. Your program, if implemented correctly and according to your algorithm, should produce the expected points value stated in each test case.
* **Step 4:** Submit a zip file with the Java file and this completed document.

During your creation, keep a couple of things in mind:

* Use comments to describe what was happening in the program.
* Choose variable names that clearly describe the data that they hold.
* Use spacing to group similar code.

## [R9] Is Your Implementation Fair?

Your Code Works… but is it fair? Now Reflect on the Tradeoffs and Conflicts

**You should never deploy real code without checking your assumptions.**

Your test cases tested your *technical* assumptions, but not *your social* assumptions.

1. Find classmates either inside or outside of the lab.
2. Run your code with them.
3. Get feedback on what worked and what didn’t.

### **Individual Work: Add your reflections to this document below.**

1. Which students are most likely to benefit from your algorithm?

Students who live far away, have low income, or have a disability benefit the most from my algorithm.

1. Which students are most likely to be forgotten by your algorithm?

Students that have bad academic performance are most likely to be forgotten by my algorithm.

*If you are interested…*

* **Optional reading:** [*What Happens When An Algorithm Cuts Your Health Care*](https://www.theverge.com/2018/3/21/17144260/healthcare-medicaid-algorithm-arkansas-cerebral-palsy) *By Colin Lecher*
* [*We created poverty. Algorithms won’t make that go away.*](https://www.theguardian.com/commentisfree/2018/may/13/we-created-poverty-algorithms-wont-make-that-go-away) *By Virginia Eubanks*