Function Name: `findDateLocation`

Inputs:
1. (*char*) First date location option
2. (*char*) Second date location option
3. (*double*) 2 x M array of online ratings for each date location
4. (*char*) 2 x 1 character column vector representing your date's preference
5. (*double*) 2 x 1 column vector of your personal rating of each location

Outputs:
1. (*char*) A string of the location where you should go on your date

Function Description:
      Valentine's Day is approaching, you have a special someone in mind to take on a date. The only problem is… you've been so busy doing your MATLAB homework that you haven't had time to figure out where to take your date! Luckily, one of your friends is an expert on cool places around Atlanta, and gives you two date location options.

      To decide which option is better, you get as many online ratings as possible on both locations, go to each location to see how much you like it, and also figure out the preferences of your date. Your function will therefore take in a 2 x M array of online ratings (scale of `1` - `10` or `0` if that particular website didn't have a rating) for both date locations, where the row represents the location and the columns are ratings from different websites. The first row corresponds to the ratings of the first location (first input) and the second row corresponds to the ratings of the second location (second input). It will also take in a 2 x 1 character column vector representing your date's preferences of each location, where a `'+'` means they like it and a `'-'` means they dislike it. The function will also take in a 2 x 1 column vector of your personal rating of the location (scale of `1` - `10`) where the first and second entries correspond to the first and second locations, respectively.

      To determine which location to take your date to, you should first consider your date's preference. If your date likes one location, and dislikes the other, then the prefered location should be chosen. If your date likes both locations, then the location with the highest average rating should be chosen. If both locations have the exact same average rating, then your personal rating should determine the location. If, however, your date dislikes both locations, then only the location with the highest average rating and with average rating greater than or equal to `7.5` should be chosen. If there is a tie between the average ratings or if neither average rating is greater than or equal to `7.5`, then your higher personal rating should determine the location.

Notes:
- `0` ratings indicate that a rating for that location wasn't available and should **not** be included in your average rating calculation.
- You will never have a tie between your personal ratings.

Function Name: `loveMeTinder`

Inputs:
1. (*double*) 1 x 2 vector of your age and maximum distance
2. (*char*) Profile of a potential date

Outputs:
1. (*char*) A string indicating a swipe to the left, a swipe to the right, or a Super Like

Function Description:

It's finally here: Singles' Awareness Day (S.A.D.). You've celebrated over the past few years with multiple pints of ice cream and reruns of *Grey's Anatomy*, but this year you've decided to try something different. Tossing your slippers aside, you've downloaded the popular "social discovery" app, Tinder, and decided that *this is the year you spend S.A.D. with someone else*. But you won't settle for just anyone; **you have standards**!

Starting with the basics, you decide to not date anyone younger than half your age plus seven years (`0.5 .* age + 7`), not to date anyone older than double your age minus seven (`(age - 7) .* 2`), and you set a maximum distance range. You then go through the profile of the potential date. A potential date's profile will be a string containing the following characteristics in this order, with each response separated by commas:

`'<Name>,<Age>,<Distance>,<College/University>,<Profile Picture Description>,<Biography>'`

You make some more qualitative observations next and decide that you aren't a big fan of selfies or quotes in biographies, but will accept up to one of those traits (but not both). If their profile picture is a selfie, the portion of their profile corresponding to the profile picture description will be `'selfie'`. The presence of quotation marks (") in their biography indicates the use of a quote. The person's actual age and distance from you will be represented by the second and third parts of the profile.

If, however, a potential date's school is `'Georgia Tech'`, then they are allowed to have both a selfie and a quote, but must still abide by the age and distance settings. You will "Super Like" their photo if they meet these conditions.

Finally, you'll call it all off if a potential date's school is u(sic)ga. No matter how well the other criteria matched, you'll go back to watching Netflix alone before dating a "dawg".

After analyzing all these criteria, you will output a simple string:

`'Swipe <left|right> on <name>''s picture'`
OR
`'Super Like <name>''s picture'`

Notes:
- A quote will start with a double quotation mark (ASCII code 34)–i.e., `'"Once...'`.
- u(sic)ga will only ever be written as `'u(sic)ga'` in the College/University portion as this is the grammatically correct spelling.
- Georgia Tech may be referred to as `'GT'`, `'Georgia Tech'`, or `'Georgia Institute of Technology'`.
- The test cases given **are not** exhaustive. Be sure to test all possible conditions!
- A swipe to the right indicates the acceptance of the potential date's profile, whereas a swipe to the left indicates rejection.
- The portions of the biography will be separated by commas, with no whitespace in between.
- Try entering the name of the (ex) Head TA into the solution function!

Function Name: `starCrossed`

Inputs:
1. (*char*) A person's birthday
2. (*char*) Another person's birthday
3. (*logical*) A 12 x 12 logical array of horoscope compatibility

Outputs:
1. (*char*) A description of the compatibility of person 1 and person 2

Function Description:

Now that you are at Georgia Tech, it is time to further your scientific knowledge. To that end, it is time to explore the one true science: Astrology*! Astrology studies character traits, predictions, and compatibility based on the position of the stars at one's birth. A common use of astrology is determining the compatibility of you and your romantic interest. There are a lot of combinations to consider, so trying to find the perfect match can be a struggle - but once again, MATLAB will come to the rescue!

Write a function `starCrossed()` that takes in 3 inputs. The first two are the birthdays of the people you want to compare. The format will be a string with the month and date separated by a space (i.e. `'December 25'`). The third input is a 12 x 12 logical array where each row and column index correspond to an astrological signs. The astrological signs and their dates are in the table below:

Sign	Dates
Aquarius	Jan. 20th - Feb. 18th
Pisces	Feb. 19th - Mar. 20th
Aries	Mar. 21st - Apr. 19th
Taurus	Apr. 20th - May 20th
Gemini	May 21st - June 20th
Cancer	June 21st - July 22nd
Leo	July 23rd - Aug. 22nd
Virgo	Aug. 23rd - Sep. 22nd
Libra	Sep. 23rd - Oct. 22nd
Scorpio	Oct. 23rd - Nov. 21st
Sagittarius	Nov. 22nd - Dec. 21st
Capricorn	Dec. 22nd - Jan. 19th

The order of the indexing in the third input corresponds to the order in this table. For example, the 1st row, 3rd column of the array is the comparison between Aquarius and Aries. Note that the table is symmetrical, meaning that this would be equal to the 3rd row, 1st column of the array. At each index is a logical corresponding to whether the two signs are compatible or not - true if they are compatible and false if they are not compatible. Your output will be a string describing the compatibility between the two signs. If they are compatible, the output string will be of the format:

```
'A <sign1> and a <sign2>? Your stars are aligned! You are destined to be
                           together.'
```

And if they are not compatible, the output string will be of the format:

```
'Your stars are crossed...a <sign1> and a <sign2> can never be together.'
```

In both of these cases, `<sign1>` and `<sign2>` should be replaced with the astrological sign of person 1 and person 2, respectively.

Notes:
- *Astrology is not the one true science**
- **Astrology is not even a science
- Don't worry about checking whether dates are valid--only valid dates will be inputted
- For the output strings, it should be exactly like the samples above; even if the signs start with a vowel, follow the format defined

Hints:
- Finding the astrological sign from the date string is a long process that you probably don't want to write out twice--this might be a good time to look into using a helper function!

Function Name: poker

Inputs:
1. (*double*) A 6 x 7 array containing the cards in each player's hand

Outputs:
1. (*char*) A string that identifies which player won and the type of hand they had

Function Description:

It's Valentine's Day! Unfortunately, all you can think about is all the money you just spent to take your date out to that fancy dinner. After thinking of all the different ways you could quickly make money, you decide on poker.* A few games of Georgia Hold'em should do the trick to get you your money back. Luckily, no knowledge of poker is necessary to play in this casino.

Each round will always consist of six players, each of whom is dealt seven cards. Suit is irrelevant in this card game. The winner will be the person who has the best hand, consisting of five out of their seven cards. The best hand is determined by the set of combinations of five cards below (in order from best to worst):

Name:	Description:	Example:
Four of a Kind	Four cards with the same value Tiebreaker: highest four card value	Winner: A,A,A,A,9 Loser: K,K,K,K,J
Full House	Three cards with the same value, and two other cards with the same value Tiebreaker: highest three card value	Winner: K,K,K,2,2 Loser: J,J,J,10,10
Straight	Five cards in a numerical order sequence Tiebreaker: Highest card end card wins	Winner: 5,6,7,8,9 Loser: 2,3,4,5,6
Three of a Kind	Three cards with the same value Tiebreaker: highest three card value	Winner: Q,Q,Q,K,2 Loser: 10,10,10,K,A
Two Pair	Two pairs of two cards with the same value Tiebreaker: highest two card value	Winner: 10,10,A,A,4 Loser: Q,Q,K,K,J
One Pair	One pair of two cards of the same rank Tiebreaker: highest two card value	Winner: J,J,5,6,10 Loser: 10,10,9,8,Q
High Card	Highest singular card	Winner: A,K,2,3,5 Loser: K,2,4,5,7

The input to the function will be an array of the cards in each person's hand. Each row will correspond to a player. The cards will contain 2-14 as possible values. The values 2-10

represent the 2-10 card values while 11 corresponds to J, 12 corresponds to Q, 13 corresponds to K, and 14 corresponds to A. You will output who has the best hand, and what they had in the format of a string saying:

`'Player <number> won with a <card combination>.'`

If the tiebreaker does not yield a single winner, or if there is not a tiebreaker, you should output:

`'Thank you for your donation to Hope and Zell Miller.'`

Notes:
- *CS1371 TAs are not allowed to endorse gambling as a way of making money.
- An Ace ONLY acts as a 14 in this game for a straight (ie. 10,J,Q,K,A), it CANNOT be used in a straight as a value of 1 (ie. A,2,3,4,5).
- Make sure to write the combination names as they appear in the table above.

Hints:
- The `mode()` and `diff()` functions may be useful to you.