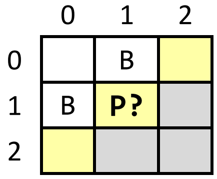
Note: Same rules apply as on previous assignments. You may NOT import anything!

P1 (10 pts) The probability that a given cell in Wumpus World (except [0, 0]) has a pit is 20%, and this independent of all other cells. You sensed a breeze in cells [0, 1] and [1, 0]. Give this observation, what is the probability that there is a pit in cell [1, 1]? You must show your work for this problem!



	0	1	2
0		B	🟨
1	B	**P?**	▨
2	🟨	▨	▨

P2 (10 pts) In file "`A7_P2_add1`", write the function `add1(array)`, which takes a list called *array* of 0's and 1's to be treated as a binary number and ***changes*** `array` to be the next binary number. It returns True if it the next binary number can be represented `array`, and False otherwise.

Below are some examples of arrays before and after the call to `add1`, as well as their return values. Note that the arrays are drawn with the lowest index on the right to make seeing the addition easier.

	3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0
before call:	1	1	0	0	1	0	1	1	0	1	1	1	1	1	1	1
after call:	1	1	0	1	1	1	0	0	1	0	0	0	0	0	0	0
return value:	True				True				True				False			

Do Problems 3 – 6 in the file "`A8_Uncertainty`".

P3 (5 pts) Write the function `probOfConfig(pit_config)` where `pit_config` is a square, 2-dimensional array with values `P` = 0 (for PIT) and `NP` = 1 (for NON-PIT). The function returns the probability of that configuration. Remember that each cell is assigned a pit with probability 0.2 and that this assignment is independent for each pit.

P4 (10 pts) Write the function `nextConfig(pit_config)`, which works like the `add1` method, but in two dimensions. This function changes the array `pit_config` to its next numeric configuration, the same way as the `add1` method, but in two-dimensions.

The picture below shows you how *nextConfig* changes *config* i to *config* $i + 1$ for all possible 2×2 arrays. The function would change *config* 15 to *config* 0 and return False. For all other calls, the function would return True.

You will receive +3 extra credit points if your function does not declare any arrays.

	0	1
0	0	0
1	0	0

0

	0	1
0	1	0
1	0	0

1

	0	1
0	0	1
1	0	0

2

	0	1
0	1	1
1	0	0

3

	0	1
0	0	0
1	1	0

4

	0	1
0	1	0
1	1	0

5

	0	1
0	0	1
1	1	0

6

	0	1
0	1	1
1	1	0

7

	0	1
0	0	0
1	0	1

8

	0	1
0	1	0
1	0	1

9

	0	1
0	0	1
1	0	1

10

	0	1
0	1	1
1	0	1

11

	0	1
0	0	0
1	1	1

12

	0	1
0	1	0
1	1	1

13

	0	1
0	0	1
1	1	1

14

	0	1
0	1	1
1	1	1

15

P5 (15 pts) Write the function `isConsitent(pit_config, obs)`, which takes a pit configuration and an observation (a square 2-D array with the same dimensions as `pit_config` containing values U (for Unvisited), B (for Breeze), and NB (for No Breeze). The function returns True if they are consistent, and False otherwise.

A pit configuration and an observation would be inconsistent if

- A cell has a pit and **ANY** adjacent cell was observed to have no breeze.
- A cell with a pit was observed to have either a breeze or no breeze, as the agent would be dead in a cell with a pit.
- A breeze is observed in a cell, all of whose neighbors have no pit (e.g., were visited).

P6 (20 pts) Write the function `getProbs(query, obs)` which takes `query`, the coordinates of a cell in the form of a list $[i, j]$, where i and j are the row and column, respectively. The function also takes an observation. The function returns a list of the form:

```
[prob_obs_and_query, prob_obs, total_prob, conditional_prob]
```

where:

- `prob_obs_and_query` = the probability of both the observation happening *and* there being a pit in the queried cell.
- `prob_obs` = the probability of the observation
- `total_prob` = the sum of the probabilites of each configuration you cycle through.
- `conditional_prob` = the conditional pobability of a pit in the query cell ***given*** the observation.

P7 (10 pts) The universe for this problem is the set of people who are tried for a crime. Suppose we know the following facts:

- 90% of people who committed a crime are found guilty.
- 99% of people who did not commit a crime are found innocent.
- 80% of the people who are tried for a crime actually did commit the crime.

Use Bayes' theorem to answer the following:

(a) What is the probability that a person who is tried and found innocent did commit the crime?
(b) What is the probability that a person who is tried and found guilty did not commit the crime?

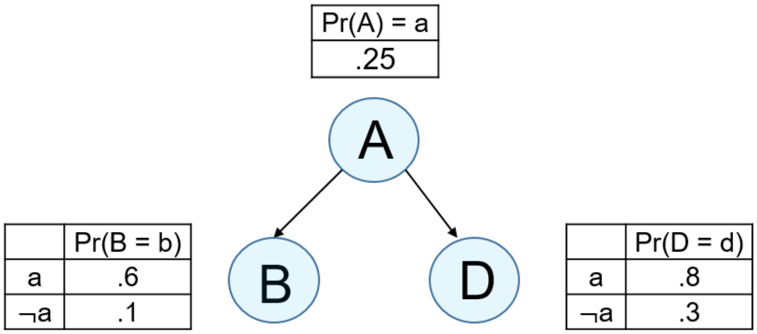
For this problem, you must ***show your work in your own handwriting!*** (You may use a simple calculator, but show all equations with numbers before evaluating, as done in class with the disease example).

P8 (10 pts) In Wumpus 2x2, we have the following observation:

	0	1
0	**B**	**U**
1	**U**	**U**

(a) Without directly computing $\Pr(obs)$, use Bayes' Theorem to compute $\Pr(pit_{0,1} \mid obs)$ and $\Pr(\neg pit_{0,1} \mid obs)$. Use the same procedure as the example in the "Model for Problem 8" file, and show your work.

(b) Use the law for the probability of the union of events to compute $\Pr(pit_{0,1} \wedge pit_{1,0} \mid obs)$



Pr(A) = a
.25

	Pr(B = b)
a	.6
¬a	.1

	Pr(D = d)
a	.8
¬a	.3

P9 (10 pts) Evaluate the following probabilities for parts (3) through (6), based on the Bayesian network above. Do not evaluate the products or sums.

1) $\Pr(a \wedge b \wedge d)$

2) $\Pr(b \wedge d \mid a)$

3) $\Pr(d)$

4) $\Pr(b \wedge d)$

5) $\Pr(a|b \wedge d)$