Name: **Worksheet 4: Conditionals** **CSCI 128**

1. Consider the scenario: you are working at a movie theater's software division and are tasked to create a program which calculates the price of tickets depending on the age of the customers and size of the group. There are two different age groups: children (Under 16) and adults. Tickets for children cost \$5 and adult tickets are \$8. There are also discounts for different sizes of groups in brackets: Groups of 1-4 people get no discount, and groups of 5 and up get a 10 percent discount on their overall admission price. Write pseudocode for a program that calculates the ticket price given the person's age and group size.

```
age = int(input())
group_size = int(input())
  is_Child = age < 16
  is_group_big = group_size > 4
   if is_Child and is_group_big:
       ticketprice = 5 · 0.9
   elif is_Child:
      ticketprice = 5 · group_size
    elif is_group_big:
       ticketPrice = 8 · 0.9 · group_size
    else:
     ticketPrice = 8 · group_size
Print(f"Ticket Price is {ticketPrice}")
```

2. Finish filling in the truth table below

a	b	not a	a or b	a and (not b)
F	F	T	F	F
F	T	T	T	F
T	F	F	T	T
T	T	F	T	F

3. If a = true, b = false, and c = true what do the following boolean expressions equal?

   (a) (a or b) and (not c or (not a or b))

   True

   (b) a or b and not c or not a or b

   False

4. (a) How would someone go about short circuiting an "and" expression?

and can be Short Circuited by having the first Condition be false.

(b) How would someone go about short circuiting an "or" expression?

Or Can be Short Circuited by having the first Condition be true.

(c) Why would this be beneficial for your code?

It can make computation faster.

5. Conveyor matrix M is a matrix of size $n \times n$, where n is an even number. The matrix consists of concentric conveyor belts moving clockwise at a speed of 1 square per second.

The conveyor matrix for $n = 2$ is a 2×2 matrix, whose cells form a cycle of length 4 clockwise. We obtain the matrix for $n = 4$ by adding an outer layer forming another conveyor, and similar for $n = 6, n = 8$, etc.

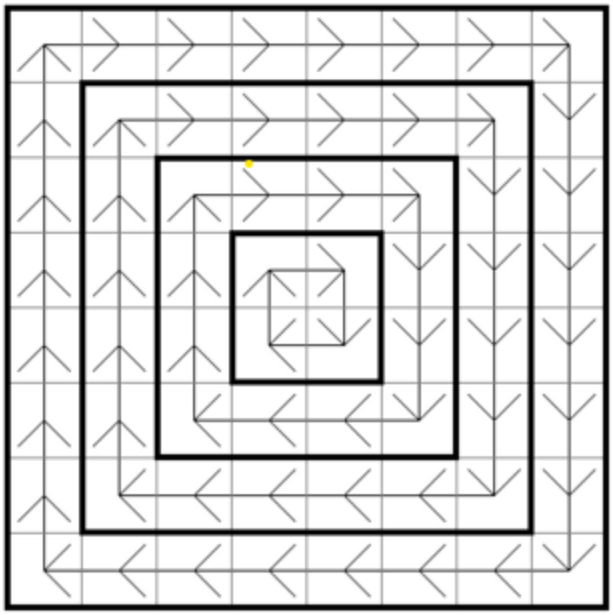

Figure 1: Conveyor matrix 8 × 8

You are standing in a cell with coordinates $x1, y1$ and would like to get to $x2, y2$. A cell has coordinates (x, y) if it is located at the intersection of the x^{th} row and the y^{th} column. As stated previously, the belts move at a speed of 1 square per second.

You can choose to move to any neighboring cell (left, right, up, or down) for a cost of 1 energy.

As an efficient (lazy?) engineer, you want to get to $x2, y2$ using the minimum amount of energy. Note we are minimizing energy, but *not* time. What is the minimum amount of energy that you must spend to get to $x2, y2$ from $x1, y1$?

(a) This problem is less a programming problem and more like a puzzle. The challenge comes in deriving a formula to solve this problem. Spend several minutes determining a method to calculate the minimum energy (Hint: try out some examples using Figure 1). Write your formula/solution below, and explain the key idea behind it using 1-2 sentences.

(b) After you are confident in your solution, implement it here using pseudocode (you shouldn't need more than 10 lines). Assume you are provided $n, x1, y1, x2, y2$. After the studio for this week opens, submit your working program to the "ConveyorMatrix" studio problem for testing and credit.