1a) We have $i = 1 \dots n$ lightbulbs
We have $j = 1 \dots m$ switches
$T_j \subseteq \{1 \dots n\}$ mapping switches to light bulbs.

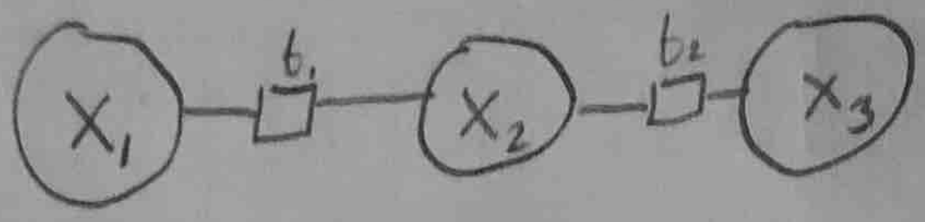
Variables = $\{button_1, button_2, \dots button_n\}$ $button_j \in \{0, 1\}$
where m is # buttons

Constraints: Doesn't have to be unary or binary

constraints: $\left(\sum_{j=1}^{m} button_j * (1(i \in T_j))\right) \% 2$ for all $i \in [1, n]$

- Have i total constraints involving all j switches
- $button_j$ has domain $\in \{0, 1\}$, if 1, switch is activated
- $\% 2$ corresponds to if lightbulb is activated an odd number of times, it will be on, constraint is satisfied.

b)



$t_1, t_2 = X_i \oplus X_j$

i) There are two consistent assignments:

$[x_1, x_2, x_3] \rightarrow [1, 0, 1], [0, 1, 0]$

ii)

w	x	Domains
1	$\emptyset$	$\{x_1, x_3, x_2\}$
1	$\{x_1 = 0\}$	$\{x_3, x_2\}$
1	$\{x_1 = 1\}$	$\{x_3, x_2\}$
1	$\{x_1 = 0, x_3 = 0\}$	$\{x_2\}$
1	$\{x_1 = 0, x_3 = 1\}$	$\{x_2\}$
1	$\{x_1 = 1, x_3 = 0\}$	$\{x_2\}$
1	$\{x_1 = 1, x_3 = 1\}$	$\{x_2\}$
0	$\{x_1 = 0, x_3 = 0, x_2 = 0\}$	$\{\}$
0	$\{x_1 = 1, x_3 = 0, x_2 = 0\}$	$\{\}$
1	$\{x_1 = 0, x_3 = 0 \; x_2 = 1\}$	$\{\}$
0	$\{x_1 = 1, x_3 = 0 \; x_2 = 1\}$	$\{\}$
0	$\{x_1 = 0, x_3 = 1 \; x_2 = 0\}$	$\{\}$
1	$\{x_1 = 1, x_3 = 1 \; x_2 = 0\}$	$\{\}$
0	$\{x_1 = 0, x_3 = 1 \; x_2 = 1\}$	$\{\}$
0	$\{x_1 = 1, x_3 = 1 \; x_2 = 1\}$	$\{\}$

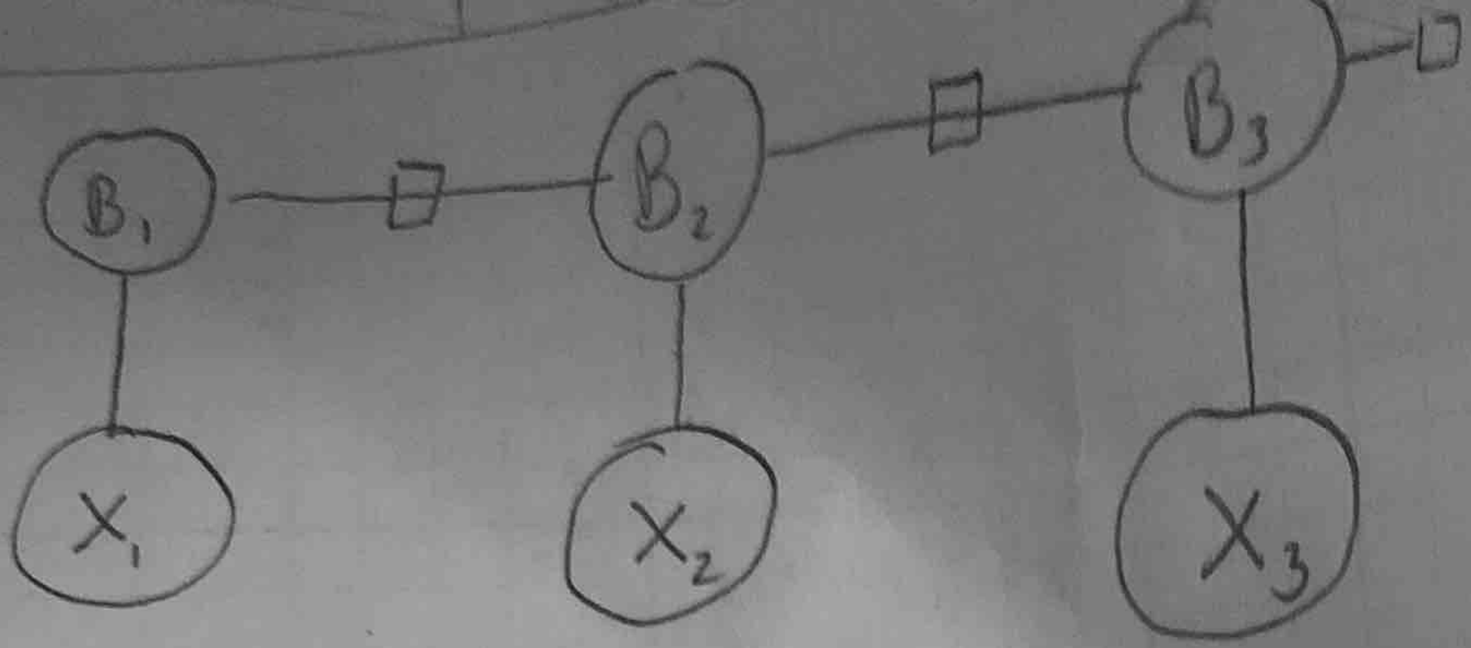
Backtrack search called 15 times, exhaustive search.

iii)

W	X	Domains	AC-Domains
1	$\emptyset$	$\{x_1\}$	—
1	$\{x_1 = 0\}$	$\{x_3, x_2\}$ (add neighbors to stack)	$[x_2 \in 1, x_3 \in 0]$
1	$\{x_1 = 0, x_2 = 1, x_3 = 0\}$	$\{\}$	—
1	$\{x_1 = 1\}$	$\{x_3, x_2\}$	$[x_2 \in 0, x_3 \in 1]$
1	$\{x_1 = 1, x_2 = 0, x_3 = 1\}$	$\{\}$	—

Bachtraching called 5 times

2a)



$X_1 + X_2 + X_3 \leq K$

- Each B_i is len 2 tuple, stores A_{i-1} and A_i

[Initialization: $[B_1[0] = 0]$

Processing: $[B_i[1] = B_i[0] + X_i]$

Consistency: $[B_i[0] = B_{i-1}[1]]$

Final Output: $1[B_3[1] \leq K]$

- B is auxiliary variable, represented $len(B_i) = 2$, first entry is value prior to X_i added in second entry after X_i added.

$B_i \in [0, 6] \times [0, 6]$

3c) Requests (Text Document):

```
# Unit limit per quarter. You can ignore this for the first
# few questions in problem 2.
minUnits 6
maxUnits 10

# These are the quarters that I need to fill. It is assumed that
# the quarters are sorted in chronological order.
register Win2018
register Spr2018
register Aut2018

# Courses I've already taken
taken CEE280
taken CEE285A
taken CEE203

# Courses that I'm requesting
request CS221 after CS106A weight 3
request CEE288 weight 4
request CEE287 in Spr2018 weight 3
request CEE385 in Aut2018 weight 2
request CEE305 in Spr2018
request CS106A weight 3
request CS106B
request CEE281
request CS108
```

```
Here's the best schedule:
Quarter         Units   Course
  Win2018       5       CS106A
  Win2018       3       CEE281
  Spr2018       4       CEE288
  Spr2018       5       CS106B
  Aut2018       4       CS221
  Aut2018       4       CS108
```

Schedule works well! Satisfies unit requirements set up, and classes are taken in the appropriate quarters. Very similiar to what my schedule is now! (Which is what I was going for)