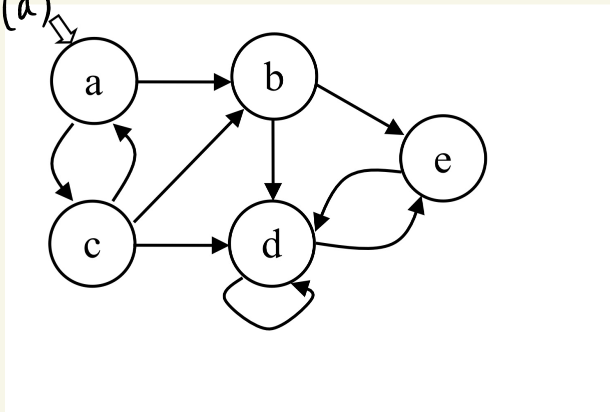
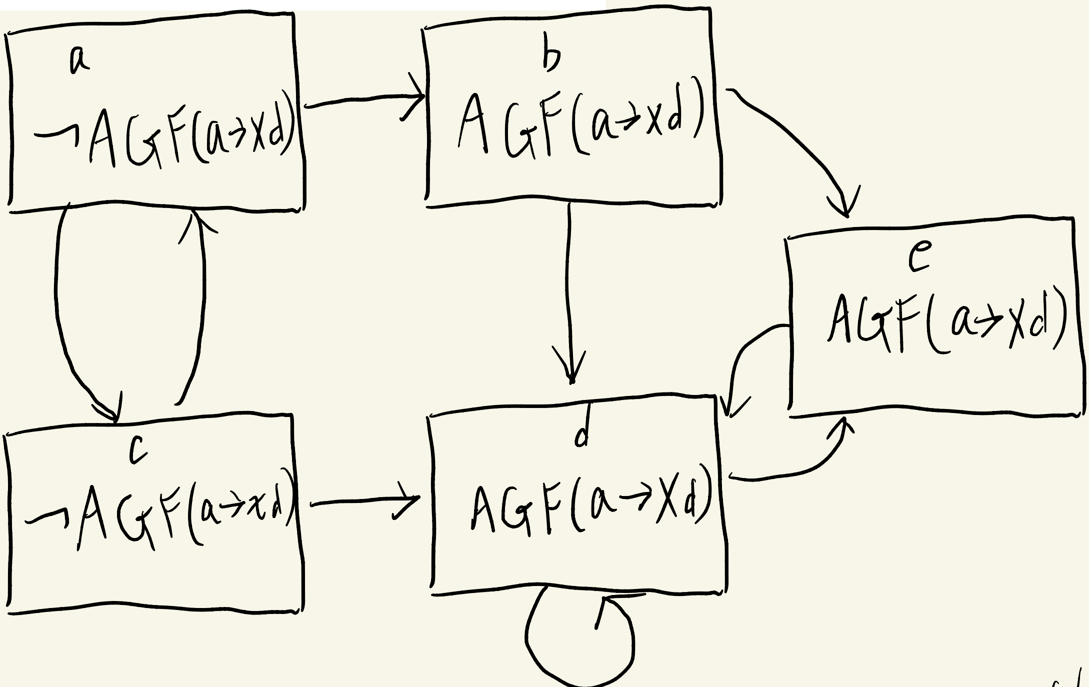
1. (a)

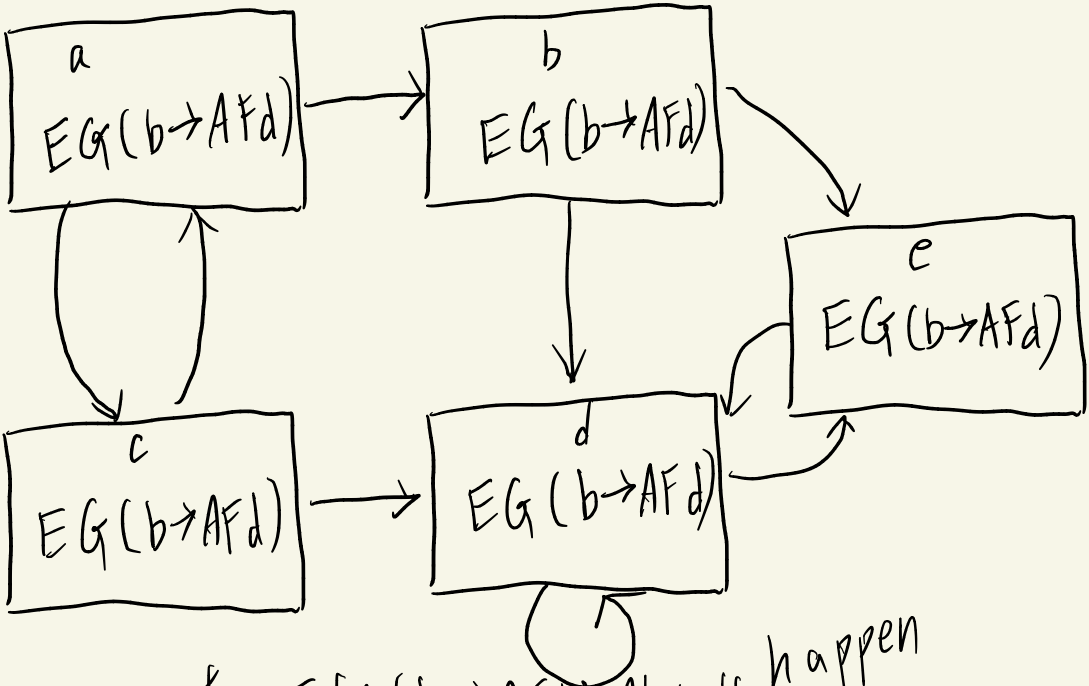


$AGF(a \to Xd)$



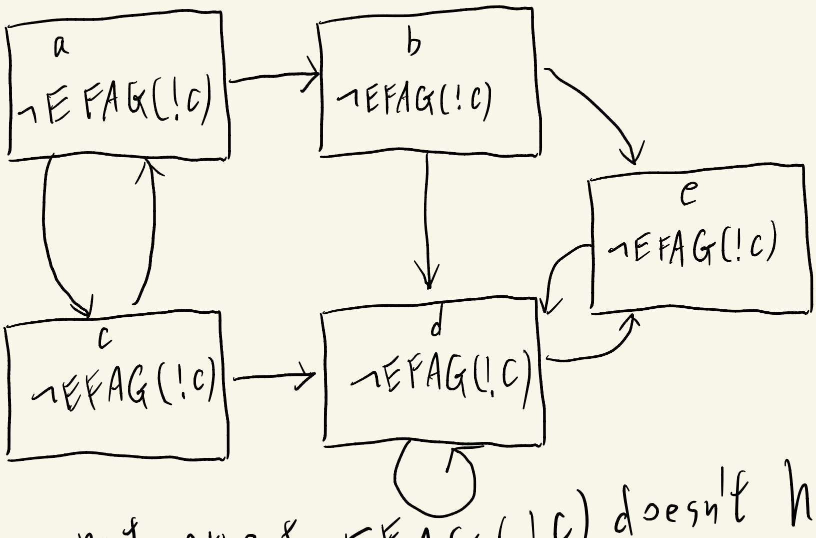
not correct, since for the path $a \to c \to a \to c \ldots$, $a \to Xd$ doesn't happen infinite often

(b) $EG(b \to AFd)$



correct, $EG(b \to AFd)$ Always happen

(c) $EFAG(!c)$

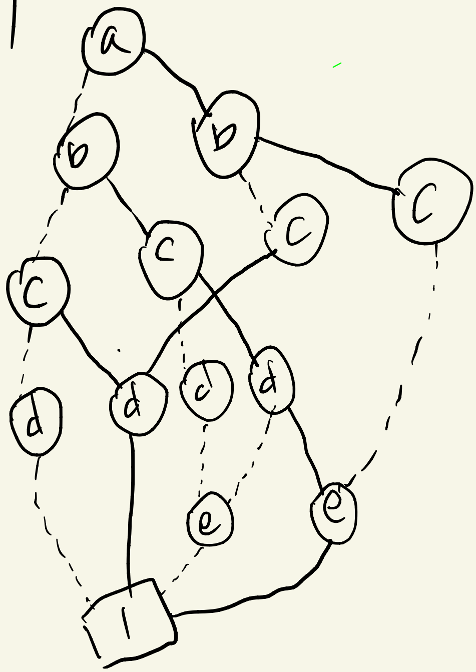


not correct, $EFAG(!c)$ doesn't hold!

2. (a) $f(0,0,0,0,0) = 1$ (c)

$f(1,0,1,0,1) = 0$

$f(0,1,0,1,0) = 0$



(b)

0	0	0	1	X
0	0	1	0	X
0	1	0	0	1
0	1	0	1	X
0	1	1	0	1
0	1	1	1	0
1	0	0	x	X
1	0	1	0	X
1	1	0	x	0
1	1	1	x	X

3.

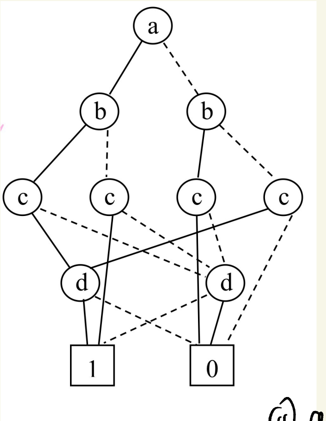
$ITE(F, 0, G)$

$= ITE(\overline{G}, 0, \overline{F})$ (symmetrical parameters)

G has smaller top variable and is complemented itself,

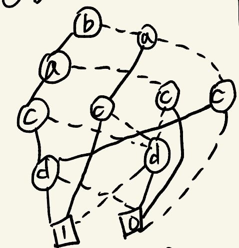
thus $\overline{G}$ is not complemented.

4.



For b:

① bacd



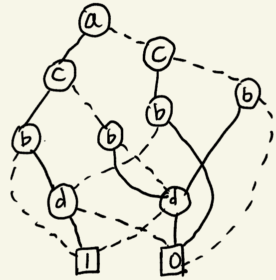
9 nodes, diff = 0

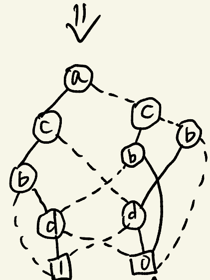
② abcd

the same as the given graph

diff = 0

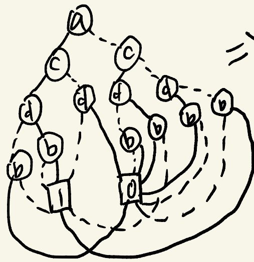
③ acbd



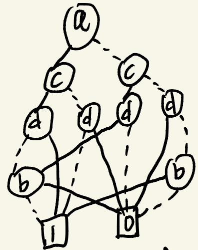


8 nodes, diff = -1

④ acdb



=>



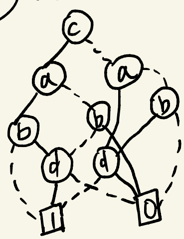
9 nodes, diff = 0

for c: ⑤ acbd

=> same as ③

diff = -1

⑥ cabd



8 nodes, diff = -1

⑦ acbd

=> same as ③

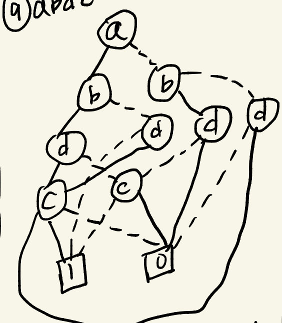
diff = -1

⑧ abcd

=> same as the given graph

diff = 0

⑨ abdc



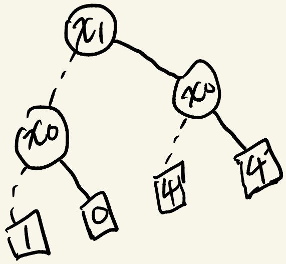
9 nodes, diff = 0

5. (a)

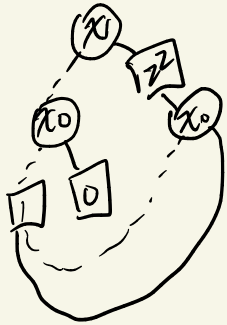
For $n=2$

x_1	x_0		
0	0	0	$0(1-x_1)(1-x_0)+$
0	1	1	$1\cdot(1-x_1)x_0+$
1	0	4	$4\cdot x_1(1-x_0)+$
1	1	9	$9\cdot x_1x_0$

MTBDD



*BMD



$$x_0 - x_1x_0 + 4x_1 - 4x_1x_0$$
$$+9x_1x_0$$
$$= x_0 + 4x_1 + 4x_1x_0$$

For $n=3$

x_2	x_1	x_0	
0	0	0	0
0	0	1	1
0	1	0	4
0	1	1	9
1	0	0	16
1	0	1	25
1	1	0	36
1	1	1	49

$(1-x_2)(1-x_1)x_0+$
$4(1-x_2)x_1(1-x_0)+$
$9\cdot(1-x_2)x_1x_0+$
$16\cdot x_2(1-x_1)(1-x_0)+$
$25\cdot x_2(1-x_1)x_0+$
$36\cdot x_2x_1(1-x_0)+$
$49x_2x_1x_0$

$x_0+4x_1+4x_1x_0$

$(1-x_1-x_2+x_2x_1)x_0+$
$(4x_1-4x_2x_1)(1-x_0)+$
$9x_1x_0-9x_2x_1x_0+$
$16x_2(1-x_0-x_1+x_1x_0)+$
$25x_2(x_0-x_1x_0)+$
$36x_2x_1-36x_2x_1x_0+$
$49x_2x_1x_0$

$(1-x_1-x_2+x_2x_1)x_0+$
$(4x_1-4x_2x_1)(1-x_0)+$
$9x_1x_0-9x_2x_1x_0+$
$16x_2(1-x_0-x_1+x_1x_0)+$
$25x_2(x_0-x_1x_0)+$
$36x_2x_1-36x_2x_1x_0+$
$49x_2x_1x_0$

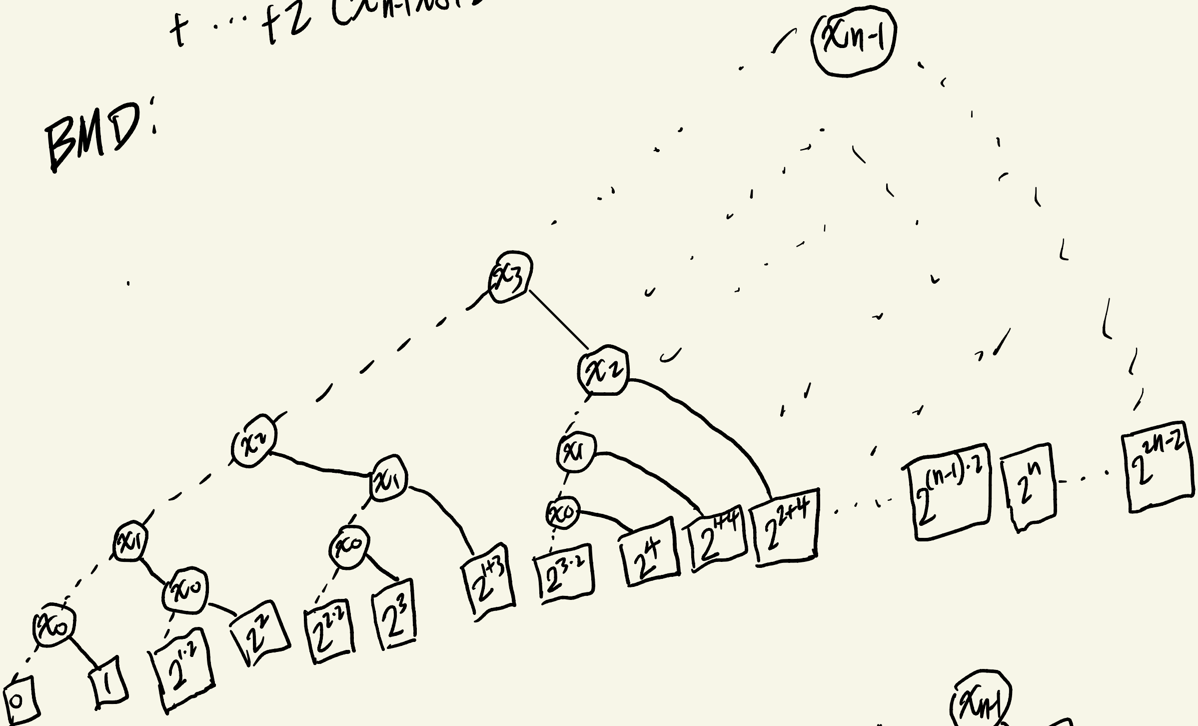
$x_0-x_1x_0-x_2x_0+x_2x_1x_0+$
$4x_1-4x_1x_0-4x_2x_1+4x_2x_1x_0+$
$9x_1x_0-9x_2x_1x_0+$
$16x_2-16x_2x_0-16x_2x_1+16x_2x_1x_0+$
$25x_2x_0-25x_2x_1x_0+36x_2x_1$
$-36x_2x_1x_0+49x_2x_1x_0$

$$(x_0+4x_1+4x_1x_0)+8x_2x_0+16x_2+16x_2x_1$$

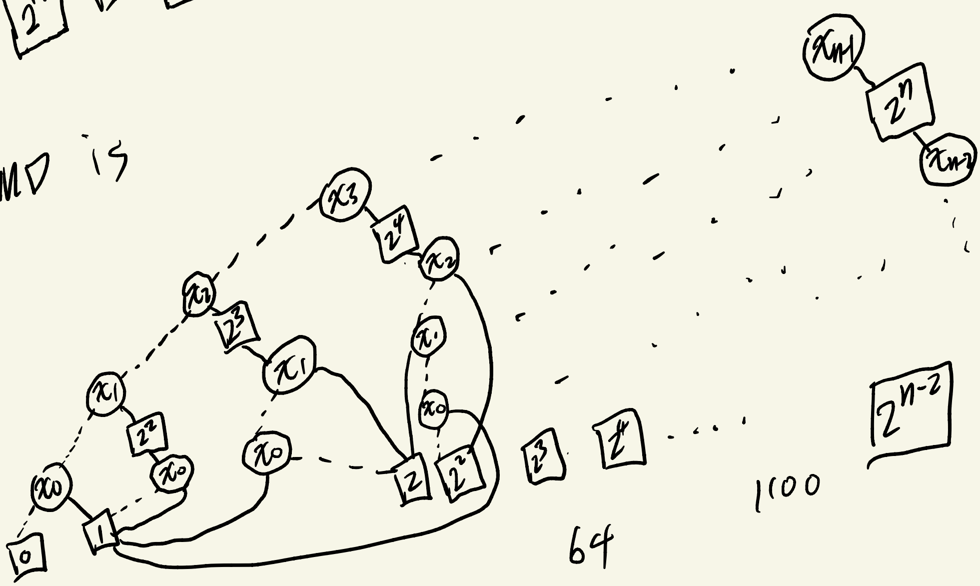
For arbitrary n:

$$(x_{n-1}x_{n-2}\cdots x_1x_0)^2 = x_0 + (2^1)^2 x_1 + (2^2)^2 x_2 + (2^3)^2 \cdot x_3 + \cdots + (2^{n-1})^2 x_{n-1} + 2^2(x_1x_0) + 2^3(x_2x_0 + 2^1 x_2x_1) + 2^4(x_3x_0 + 2^1 x_3x_1 + 2^2 x_3x_2) + \cdots + 2^n(x_{n-1}x_0 + 2^1 x_{n-1}x_1 + \cdots + 2^{n-2} x_{n-1}x_{n-2})$$

BMD:



*BMD is



(b) pseudo code:

```
input Y
output X = (x_{n-1} x_{n-2} ... x_1 x_0)_2  (a binary number)
int k = 0, value = 0, tmp = 1
while True:
    from x_k, go pos-edge to x_{k+1} and go neg-edge until reach terminal
        tmp = tmp x (the value of edge)
        // tmp = 2^{2·k}
    if tmp >= Y:
        n = k+1 // The MSB of X is k+1
        x_{n-1} = 1
        break
    else:
        value = tmp
        k = k+1
X = [x_{n-1} = 1, x_{n-2} = 0, x_{n-3} = 0, ... x_1 = 0, x_0 = 0] // To record the values of the bits
// We now only consider the *BMD's top variable as x_{n-1}, as computed n-1
for i = n-2 down to 0:
    for all path with x_i = 1 and x_{j<i} = 0:
        tmp = 1
        for the edge_value of the path:
            tmp = tmp x edge_value
    if value + tmp <= Y:
        value = value + Y
        Arr[x_i] = 1
return X (turn the dictionery into a binary number)
```

Complexity:

Since each edge is traversed only once and the Edge number is at most $2\times(\#\text{Nodes}) = \cancel{2}\times\frac{n(n+1)}{\cancel{2}} = n^2+n$

Thus my algorithm is $O(n^2)$