TP 3

2. Programme TANTQUE pour Ackermann

$A(i,x) := B_i(x)$

si $i=0$: $A(i,x) = A(0,x) = B_0(x) = x+1$

si $i>0$: si $x=0$,
$$\begin{aligned} A(i,x) &= B_i(x) \\ &= B_i(0) \\ &= B_{i-1}(1) \\ &= A(i-1,1) \end{aligned}$$

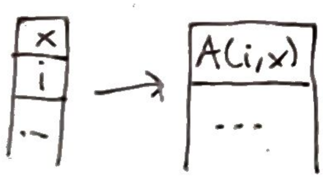
si $x>0$,
$$\begin{aligned} A(i,x) &= B_i(x) \\ &= B_{i-1}^{\langle x+1 \rangle}(1) \\ &= B_{i-1}(B_{i-1}^{\langle x \rangle}(1)) \\ &= B_{i-1}(B_i(x-1)) \\ &= A(i-1, A(i, x-1)) \end{aligned}$$

$$A(i,x) = \begin{cases} x+1 & \text{si } i=0 \\ A(i-1,1) & \text{si } i>0 \text{ et } x=0 \\ A(i-1, A(i,x-1)) & \text{sinon} \end{cases}$$

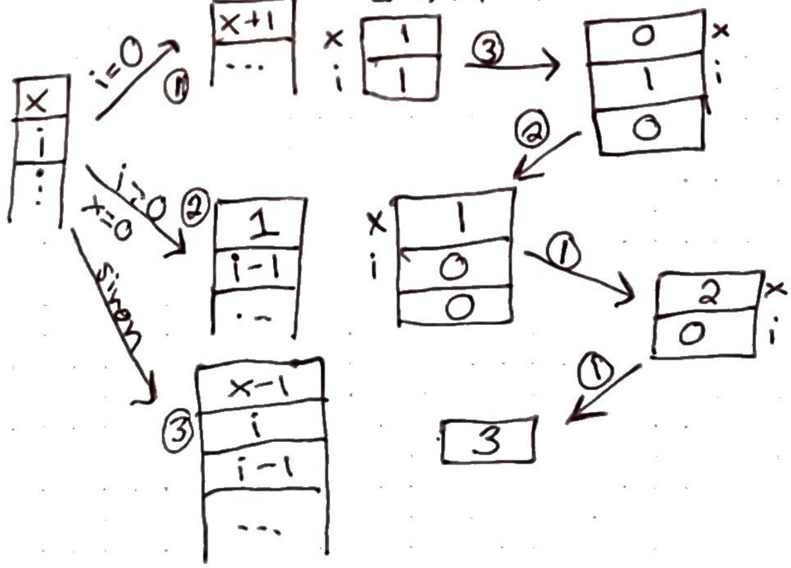
Ex. : $A(\overset{i}{1}, \overset{x}{1}) = A(0, A(\overset{①}{1}, 0)) = A(0,2) = 3$

① $A(1,0) = A(0,1) = 2$

Avec une pile :



Ex. : A(1, 1)



PseudoTANTQUE:

```
A(i, x):
  empile(i)
  empile(x)
  tant que la pile contient 2 elem ou plus
        x' ← dépile
        i' ← dépile
        si i'=0 alors
           empiler(x'+1)
        si i'>0 et x'=0 alors
           empile(i'-1)
           empile(1)
        sinon
           empile(i'-1)
           empile(i')
           empile(x'-1)
  r0 ← dépile
```

TANTQUE: Pile: P hauteur de la pile: t

```
  P ← 1
  inc(t) : empile(i)
  P ← TABLASS(P, t, i)
  inc(t) empile(x)
  P ← TABLASS(P, t, x)
  tant que t≠1 faire
     x' ← TABLVAL(P, t) : x' ← dépile
     i' ← TABLVAL(P, dec(t)) : i' ← dépile
     si ET[PG?(i', 0), x'=0] alors :          si i'>0 et x'=0 alors → faire une macro
     si ET[PG?(i', 0), PG?(x', 0)] alors :    ← sinon
```

3. Machine de Turing pour 0^n1^n

On a vu une MT M qui accepte $\{0^n1^n \mid n \geq 0\}$

	⊔	⊔	⊔	0	0	0	1	1	1	⊔	⊔

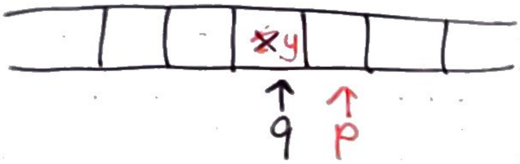
$M = (Q, \Sigma, \Gamma, \delta, q_0, F)$ $\Sigma = \{0, 1\}$

$Q = \{q_0, q_1, q_2, q_3, q_A, q_R\}$ $\Gamma = \{⊔, 0, 1, Z, U\}$

$F = \{q_A, q_R\}$

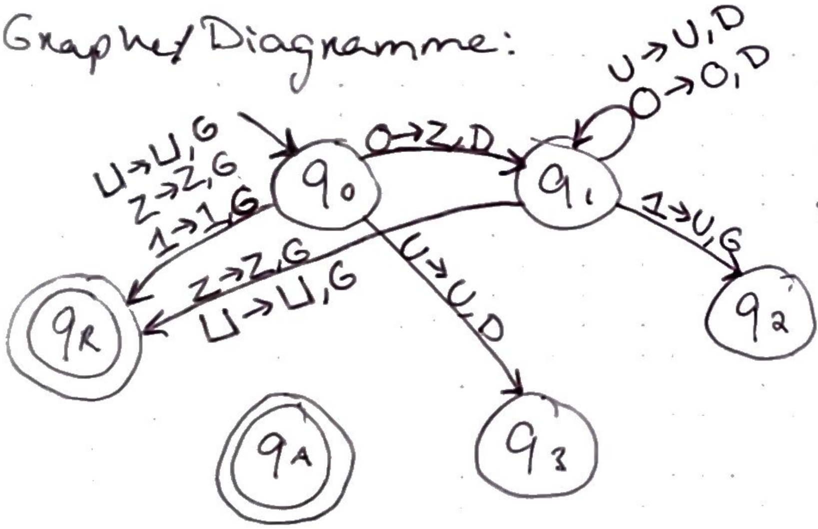
acceptant refusant

$\delta : Q \times \Gamma \to Q \times \Gamma \times \{G, D\}$

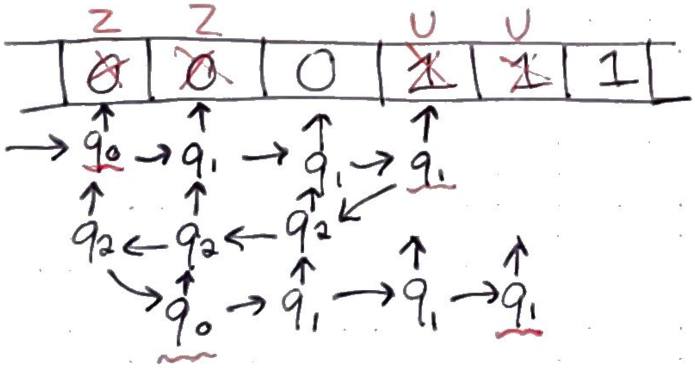


$\delta(q, x) = (p, y, D)$

Graphe/Diagramme:



Reste à faire la même chose avec q_2 et q_3.



4. Machine de Turing pour calculer n+1

Ex.: 011 → 100 → 1000 → 1001

3 4 8 9

↓

111 → 1000

7 8

Idée: On part de la droite, on change les 1 en 0 jusqu'à ce qu'on atteigne un 0 qu'on change en 1.

$M = (Q, \Sigma, \Gamma, \delta, q_0, F)$

Au départ, on rajoute un 0 à gauche.

$Q = \{q_0, q_1, q_2, q_3, q_F\}$ ← pcq calculant

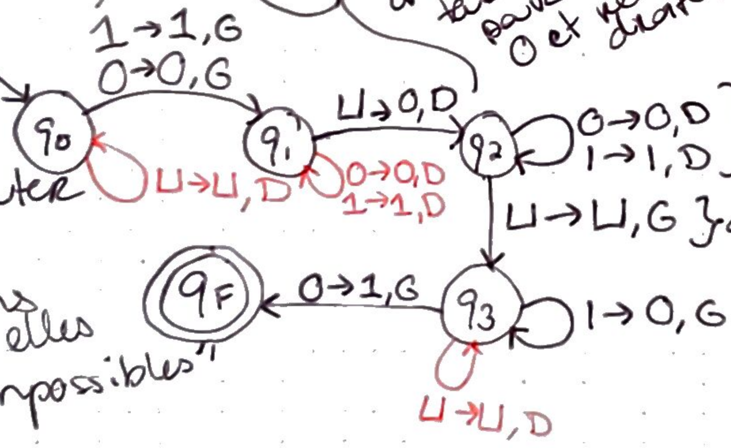
$\Sigma = \{0, 1\}$

$\Gamma = \{\sqcup, 0, 1, \quad \}$

$F = \{q_F\}$

La première chose est de se tasser à gauche pour mettre un 0 et retourner à droite.

δ:



} on se rend au bout du mot à droite (vide)

} on va à gauche au LSB

Faut rajouter les autres transitions même si elles sont "impossibles"