



# PUSHDOWN AUTOMATA & TURING MACHINE

FIRDAUS SOLIHIN



# PUSHDOWN AUTOMATA (PDA)

# [ PDA ]

---

## PUSHDOWN AUTOMATA

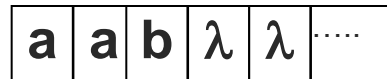
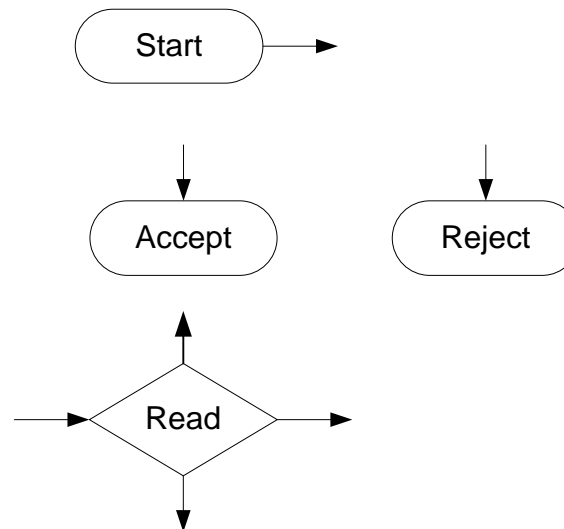
- Mesin Automata dari Non regular Language (bahasa bebas konteks)

# Komponen PDA

- Posisi awal
- Posisi Akhir

- Transisi

- Tape (penyimpan karakter input)

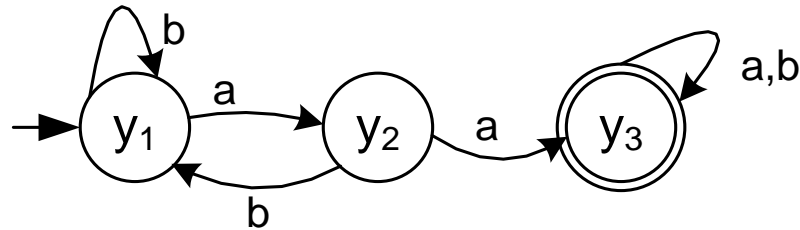


# [Aturan Penerimaan PDA]

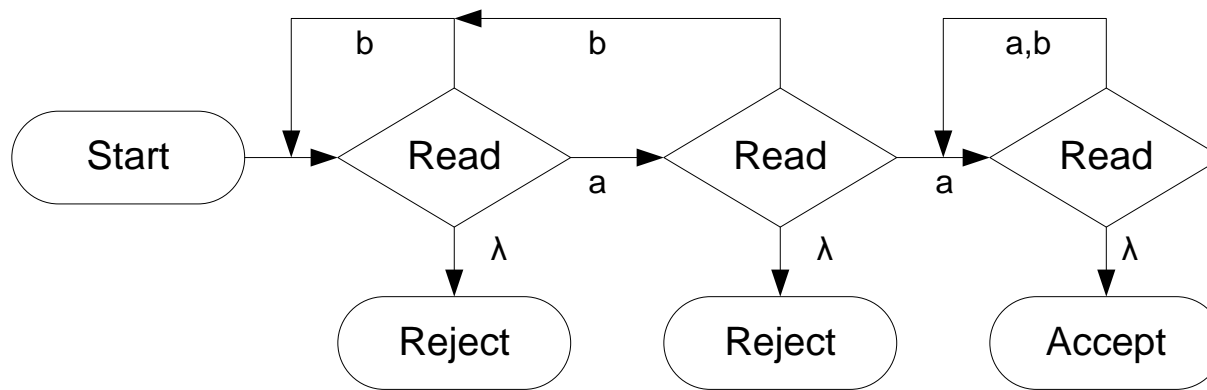
- Suatu proses pada PDA/Input string yang tersimpan pada TAPE akan diterima jika:
  - proses itu diawali dari posisi start dan
  - berakhir pada posisi accept

# Contoh PDA (1)

## ■ FA

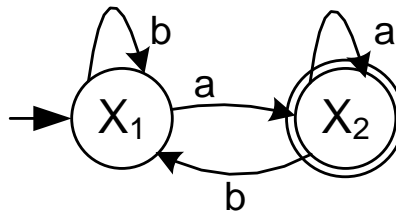


## ■ PDA

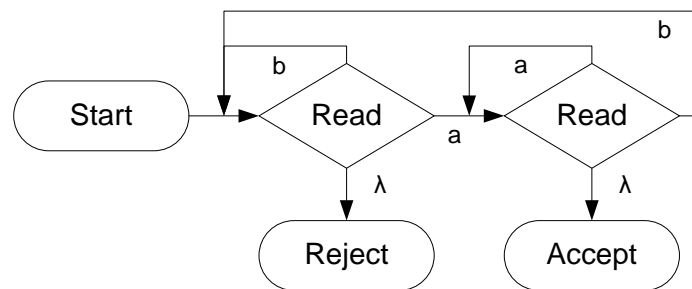


# [ Contoh PDA (2) ]

## ■ FA



## ■ PDA



# [ PDS ]

---

## PUSHDOWN STACK

- PDS adalah pengembangan dari PDA dengan menambahkan proses memasukkan nilai (Push), mengambil nilai (Pop) dari media penyimpanan (Stack)



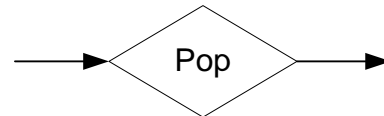
# [Komponen PDS]

- Semua komponen PDA, ditambah:

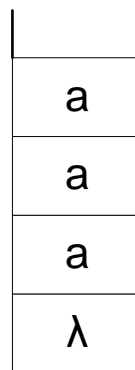
- Push



- Pop



- Stack

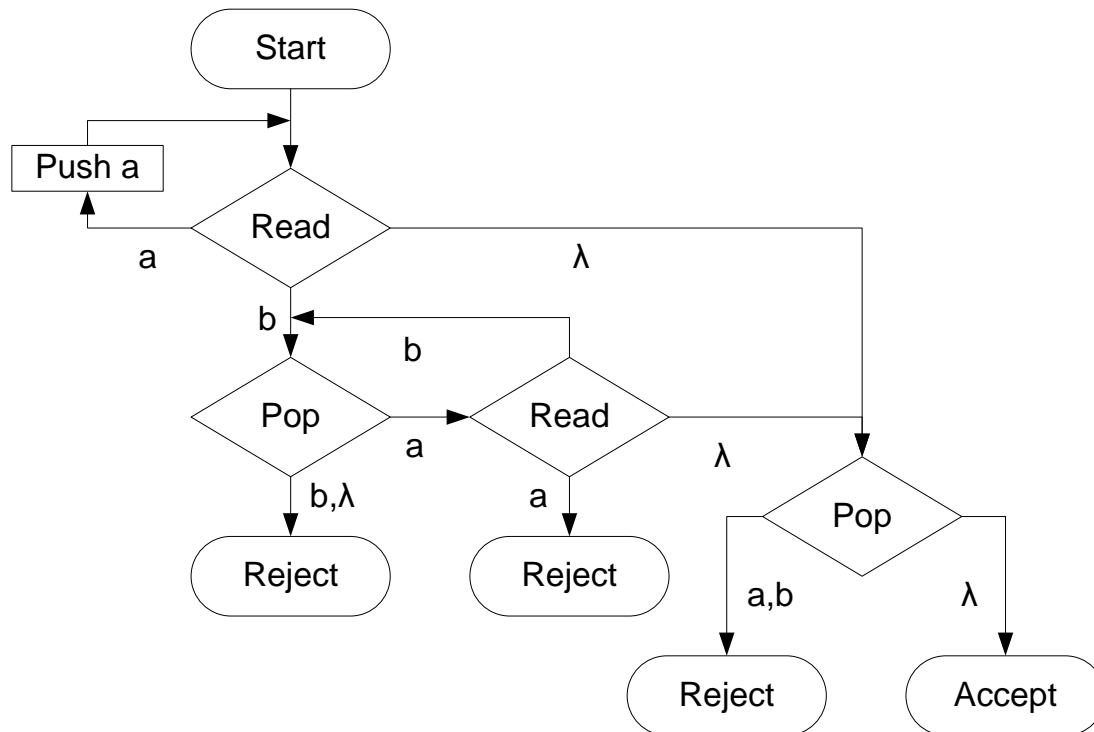


# [Aturan Penerimaan PDS]

- Suatu proses pada PDS/Input string yang tersimpan pada TAPE akan diterima jika:
  - proses itu diawali dari posisi start dan
  - berakhir pada posisi accept serta
  - stack dalam keadaan kosong

# Contoh PDS

- PDS yang mewakili bahasa  $L = a^n b^n$  dimana  $n=0,1,2,3, \dots$





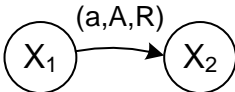
# [Test Input PDS]

- Periksa String dibawah ini
  - ab
  - abab
  - abba
  - aabb
  - aaabbb



# TURING MACHINE (TM)

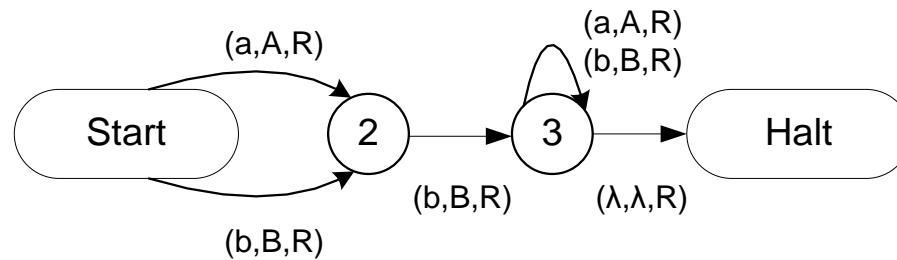
# [Komponen TM]

- Start State 
- Halt State 
- Tape (penyimpanan string input)
- Head (pembaca karakter pada tape yang dapat diatur Right [R] dan Left [L])
- Rule 
  - a = karakter input yang sedang dibaca
  - A = karakter output yang dicetak
  - R = arah gerakan Head

# Contoh TM (1)

■ RE:  $(a+b)b(a+b)^*$

■ TM

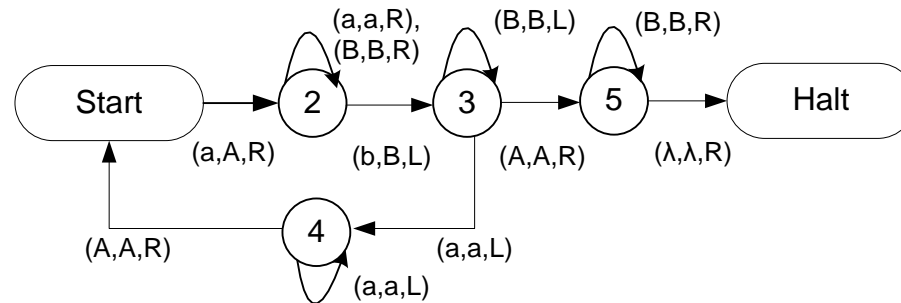


■ Input a b a  $\lambda$

- a b a  $\lambda$
- A b a  $\lambda$
- A B a  $\lambda$
- A B A  $\lambda$
- A B A  $\lambda$

# [ Contoh TM (2) ]

- $L = a^n b^n, n=1,2,3,\dots$
- TM



Input = a a b b λ

1. Start	<u>a</u> a b b	8. 2	A A B <u>b</u>
2. 2	A <u>a</u> b b	9. 3	A A <u>B</u> B
3. 2	A a <u>b</u> b	10. 3	A <u>A</u> B B
4. 3	A <u>a</u> B b	11. 5	A A <u>B</u> B
5. 4	<u>A</u> a B b	12. 5	A A B <u>B</u>
6. Start	A <u>a</u> B b	13. 5	A A B B <u>λ</u>
7. 2	A A <u>B</u> b	14. Halt	A A B B λ



[ INPUT = a a a b b b ]

1. Start a a a b b b
2. 2 A a a b b b
3. 2 A a a b b b
4. 2 A a a b b b
5. 3 A a a B b b
6. 4 A a a B b b
7. Start A a a B b b
8. 2 A A a B b b