Deterministic and Non-deterministic PDA ①

Let $M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$ be a PDA. (Lecture-23)

- The PDA is deterministic if
  (1) $\delta(q, a, z)$ has only one element
  (2) If $\delta(q, \epsilon, z)$ is non-empty, then $\delta(q, a, z)$ should be empty.
- Both conditions should be satisfied
  - if one condition fails - PDA is non-deterministic

Q/ Is the PDA to accept the language $L(M) = \{w \mid w \in (a+b)^*$ and $n_a(w) = n_b(w)\}$ deterministic.

Sol First make the pda to accept the lang given.

Step 1 :→

$\delta(q_0, a, z_0) = (q_0, az_0)$

$\delta(q_0, b, z_0) = (q_0, bz_0)$

Step 2.

$\delta(q_0, a, a) = (q_0, aa)$

$\delta(q_0, b, b) = (q_0, bb)$

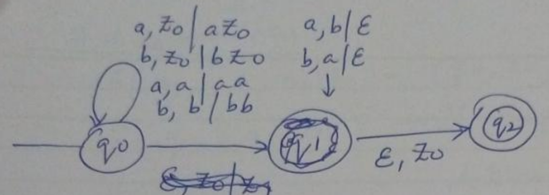
$\delta(q_0, a, b) = (q_1, \epsilon)$

$\delta(q_0, b, a) = (q_1, \epsilon)$.

- push first scanned input on the stack.
- from the next find if it is diff from the one on stack, pop one, otherwise push. continue till the string is over or stack is em

Step 3

$\delta(q_1, \varepsilon, Z_0) = (q_2, Z_0)$



Conditions for checking determinism

① $\delta(q, a, Z)$ has only one component

$\delta(q_1, \varepsilon, Z_0) = (q_2, \varepsilon)$

↑ empty.

$\delta(q_1, a, Z_0)$ is not defined.

⇓ deterministic.

Q) Balanced parenthesis

$\delta(q_0, (, Z_0) = (q_1, (Z_0)$

$\delta(q_0, [, Z_0) = (q_1, [Z_0)$

$\delta(q_1, (, () = (q_1, (()$

$\delta(q_1, (, [) = (q_1, ([)$

$\delta(q_1, [, () = (q_1, [()$

$\delta(q_1,), () = (q_1, \varepsilon)$

$\delta(q_1,], [) = (q_1, \varepsilon)$

$\delta(q_1, \varepsilon, Z_0) = (q_0, Z_0)$

First condition

$\delta(q, a, z)$ — should have only one component

2nd condition

$\delta(q_1, \varepsilon, z_0) = (q_0, z_0)$

$\delta(q_1, a, z_0)$ will not go to (q_0, z_0)

$\Downarrow$ deterministic

Q/ Is the PDA to accept $L = \{ w w^R \mid w \in (a+b)^* \}$ deterministic

X

Q $L = \{ w \mid w \in (a+b)^* \; n_a(w) > n_b(w) \}$ is deterministic?

$\delta(q_0, a, z_0) = (q_0, a z_0)$

$\delta(q_0, b, z_0) = (q_0, b z_0)$

$\delta(q_0, a, a) = (q_0, aa)$

$\delta(q_0, b, b) = (q_0, bb)$

$\delta(q_0, a, b) = (q_0, \varepsilon)$

$\delta(q_0, b, a) = (q_0, \varepsilon)$

$\delta(q_0, \varepsilon, a) = (q_1, a)$

Non-deterministic

First condition

$\delta(q, a, z_0)$ should have one component

2nd condition

$\delta(q_0, \varepsilon, a) = (q_1, a)$ — defined

defined

$\Downarrow$

non-deterministic

But $\delta(q_0, a, a) = (q_0, aa)$

and $\delta(q_0, b, a) = (q_0, \varepsilon)$

top of stack was a

CFG to PDA

- getting a PDA from CFG is possible when CFG is in GNF.
- To obtain PDA from CFG, first obtain the grammar in GNF. steps are :

1) Convert grammar into ~~CFG~~ GNF.
2) Let q_0 and Z_0 be the start state and initial symbol in stack.

   Without consuming any input, push the start symbol S on to the stack and change state to q_1. Transition can be

   $$\delta(q_0, \epsilon, Z_0) = (q_1, S Z_0)$$

3. For each production of the form

   $$A \rightarrow a\alpha$$

   introduce the transition

   $$\delta(q_1, a, A) = (q_1, \alpha)$$

4. Finally in state q_1, without consuming any input, change the state to q_f which is an accepting state. The transition for this can be of the form

   $$\delta(q_1, \epsilon, Z_0) = (q_f, Z_0)$$

Obtain PDA for the grammar below (5)

$S \to aABC$

$A \to aB \mid a$

$B \to bA \mid b$

$C \to a$

Sol: Let q_0, Z_0 be start sym and stack initial sym.

Step 1 Push start symbol on to the stack and change state to q_1.

$$\delta(q_0, \epsilon, Z_0) = (q_1, SZ_0) \quad -①$$

Step 2 For each production of the form $A \to a\alpha$, introduce the transition

$\delta(q_1, a, A) = (q_1, \alpha)$. This can be shown

Production	Transition	
$S \to aABC$	$\delta(q_1, a, S) = (q_1, ABC)$	– 2
$A \to aB$	$\delta(q_1, a, A) = (q_1, B)$	– 3
$A \to a$	$\delta(q_1, a, A) = (q_1, \epsilon)$	– 4
$B \to bA$	$\delta(q_1, b, B) = (q_1, B)$	– 5
$B \to b$	$\delta(q_1, b, B) = (q_1, \epsilon)$	– 6
$C \to a$	$\delta(q_1, a, C) = (q_1, \epsilon)$	– 7

Step 3 Finally in state q_1, without consuming any input, change the state to q_f which is an accepting state.

$$\delta(q_1, \epsilon, Z_0) = (q_f, Z_0) \quad - 8$$

– Terminals of grammar G will be the input symbols in the PDA and non-terminals will be the stack symbol in PDA.

$S \to aABC$
$\to aaBBC$
$\to aabBC$
$\to aabbC$
$\to aabba.$ check if this is accepted by the PDA. using

$(q_0, aabba, Z_0) \vdash (q_1, aabba, SZ_0)$ — 1
$\vdash (q_1, abba, ABCZ_0)$ — 2
$\vdash (q_1, bba, BBCZ_0)$ — 3
$\vdash (q_1, ba, BCZ_0)$ — ~~6~~
$\vdash (q_1, a, CZ_0)$ — 6
$\vdash (q_1, \varepsilon, Z_0)$ — 7
$\vdash (q_f, \varepsilon, Z_0)$ — 8.
is accepted

Q) $S \to aABB \mid aAA$
$A \to aBB \mid a$
$B \to bBB \mid A$
$C \to a.$

Sol. To obtain a PDA from CFG, the G must be in GNF. $B \to A$ is not in GNF.

$B \to bBB \mid aBB \mid a$

$\Downarrow$

$S \to aABB \mid aAA$
$A \to aBB \mid a$
$B \to bBB \mid aBB \mid a$
$C \to a$

7

Let q_0 be the start state and Z_0 be the initial symbol on the stack.

Step 1 :→ Push the start symbol S on to the stack and change state to q_1.

$$\delta(q_0, \varepsilon, Z_0) = (q_1, SZ_0)$$

Step 2 :→ For each production $A \to a\alpha$. introduce the transition

$$\delta(q_1, a, A) = (q_1, \alpha)$$

Production	Transition
$S \to aABB$	$\delta(q_1, a, S) = (q_1, ABB)$
$S \to aAA$	$\delta(q_1, a, S) = (q_1, AA)$
$A \to aBB$	$\delta(q_1, a, A) = (q_1, BB)$
$A \to a$	$\delta(q_1, a, A) = (q_1, \varepsilon)$
$B \to bBB$	$\delta(q_1, b, B) = (q_1, BB)$
$B \to aBB$	$\delta(q_1, a, B) = (q_1, BB)$
$B \to a$	$\delta(q_1, a, B) = (q_1, \varepsilon)$
$C \to a$	$\delta(q_1, a, C) = (q_1, \varepsilon)$

$$\delta(q_1, \varepsilon, Z_0) = (q_0, Z_0)$$

Write everything in M.