

Figure: LR(0) Parse Table

• Action part এ সব terminals থাকবে

Goto part এ সব non terminals থাকবে.

Action এ input buffer থেকে stack এ ঢুকানো থাকবে,

• Table দেখে decision নিবে. (অন্যটা go to fill up)

4 ধরনের action আছে-

- Shift
- Reduce
- Accept
- Error

$E \rightarrow E + T$	_____	(1)
$E \rightarrow T$	_____	(2)
$T \rightarrow T * F$	_____	(3)
$T \rightarrow F$	_____	(4)
$F \rightarrow (E)$	_____	(5)
$F \rightarrow id$	_____	(6)

❗ 0 state ওয়া \$ দেখে decision নিতে বসলে error উঠবে,

↳ ans is error.

ফাঁকা box মানে error.

RA মানে reduced by rule no 4

Reduction এ action সবার জন্য উঠবে

• Accept বুঝিয়ে \$ এর নিচে বসবে,

• $E \rightarrow E'$ হয়ে item set এ completely পূর্ণা- শেষ হয় এ item set এর \$ এর নিচে accept লিখবে.

• Augmented rule বাদে অন্য কোনো grammar completely পূর্ণা শেষ হলে. reduce,

↳ Reduced by কত number rules.

↳ এতটুকু box এর under এ reduced by কত লিখবে.

Table দেখে decision নিবে,

⇒ 0 state এ E দেখে I1 এ থাকি,

⇒ 0 state এ (দেখে I4 এ থাকি ; ১০ এখানে shift

হবে. (দেখে shift করে I4 এ থাকি,


১৭ মানে shift to four


■ ଏହାକୁ କି *successfully parsing* କରାଯାଏ?

⇒ *parser* 21, as item set 2 1 + *parser* 57 କରାଯାଏ
R2 କରାଯାଏ *am confused*, so *successfully parsing* କରାଯାଏ
parser 21,

Conflict ହଟାଇ *parser* *input string successfully*
parse କରାଯାଏ *parser* 21.

Given grammar 21 LR(0) grammar 21, LR(0) grammar 21
parse table 1 *error* *error* 21,

2 *conflict* *error* 21,  ଏକ *item* 1 *error* 21
21 *shift* & *reduce* *error* 21:
21: 57/R2

• *Shift Reduce Conflict*
• *Reduce Reduce Conflict*  ଏକ *item* 1 *error* 21
21 *reduce* *error* 21:
21: R1/R2.

First set, Follow set, SLR(1) *parser* 1 *error* 21,
non terminal 1 *first set* 1 *follow*
set 1 *error* 21.