

به نام خدا

فاطمه زهرا بخشنده

$\text{Addr} \rightarrow \text{Name} @ \text{id} . \text{Name}$

$\text{Name} \rightarrow \text{id Name}'$

$\text{Name}' \rightarrow \epsilon \mid . \text{id Name}'$

$\text{id} \rightarrow \text{'ghazal'} \mid 98522157$

$\text{First}(\text{Name}') = \{., \epsilon\}$

$\text{Follow}(\text{Name}') = \text{Follow}(\text{Name}) = \{@\}$ U $\text{follow}(\text{Addr}) = \{@, \$\}$

این گرامر left recursion ندارد.

Left factoring نیز ندارد.

فقط 'Name' nullable است که مجموعه first و follow آن با هم اشتراک ندارند.

برای اینکه کاملاً مطمئن باشیم LL1 است باید parse table آن را تشکیل دهیم. ابتدا first و follow همه را بدست می آوریم. (چون سوال بعدش عوض شد دیگه id رو از terminal ها پاک نکردم چون گفتید مشکلی نداره)

	First	Follow
Addr	{id, g, 9}	{\$}
Name	{id, g, 9}	{@, \$}
Name'	{., ε}	{@, \$}
id	{g, 9}	{., @, \$}

سپس parse table آن را تشکیل می دهیم.

چون بقیه ترمینال ها (یعنی h و a و z و l و 8 و 5 و 2 و 1 و 7) در هیچ مجموعه first ای نبودند آن ها را در جدول parse tree نیاردم.

	@	id	.	\$	ghazal	98522157
Addr		Name @ id . Name			Name @ id . Name	Name @ id . Name
Name		id Name'			id Name'	id Name'
Name'	ε		. id Name'	ε		
id					ghazal	98522157

در همه خانه های جدول تنها یک rule وجود دارد پس مطمئن شدیم که گرامر LL1 است.

input = [ghazal@98522157.ghazal](#) => Name = ghazal, id = 98522157

Stack	Input	Action
Addr	ghazal@98522157.ghazal	Addr -> Name @ id . Name
Name . id @ Name	ghazal@98522157.ghazal	Name -> id Name'
Name . id @ Name' id	ghazal@98522157.ghazal	id → ghazal
Name . id @ Name' ghazal	ghazal@98522157.ghazal	Delete
Name . id @ Name'	@98522157.ghazal	Name' -> ε
Name . id @	@98522157.ghazal	Delete
Name . id	98522157.ghazal	id -> 98522157
Name . 98522157	98522157.ghazal	Delete
Name .	.ghazal	Delete
Name	ghazal	Name -> id Name'
Name' id	ghazal	id -> ghazal
Name' ghazal	ghazal	Delete
Name'	ε	Name' -> ε
ε	ε	Delete