

# Muhammad Maaz Tariq

## 2019342

### CS-424: Compiler Construction

### Assignment 2

#### Sets Of LR(1) items:

Going step by step here and showing all working. Firstly, augmenting the grammar by adding a new start symbol and a new production.

$S' \rightarrow E$

Converting the grammar to an LR(0) grammar by adding a new item for each production.

$S' \rightarrow .E$   
 $E \rightarrow .T-E$   
 $E \rightarrow .T$   
 $T \rightarrow .F \times T$   
 $T \rightarrow .F$   
 $F \rightarrow .id$

Computing the closure of each LR(0) item.

$\text{Closure}(\{S' \rightarrow .E\}) = \{S' \rightarrow .E\}$   
 $\text{Closure}(\{E \rightarrow .T-E\}) = \{E \rightarrow .T-E, T \rightarrow .F \times T, T \rightarrow .F, F \rightarrow .id\}$   
 $\text{Closure}(\{E \rightarrow .T\}) = \{E \rightarrow .T, T \rightarrow .F \times T, T \rightarrow .F, F \rightarrow .id\}$   
 $\text{Closure}(\{T \rightarrow .F \times T\}) = \{T \rightarrow .F \times T, F \rightarrow .id\}$   
 $\text{Closure}(\{T \rightarrow .F\}) = \{T \rightarrow .F, F \rightarrow .id\}$   
 $\text{Closure}(\{F \rightarrow .id\}) = \{F \rightarrow .id\}$

Computing the GOTO sets for each LR(0) item and each grammar symbol.

$\text{GOTO}(\{S' \rightarrow .E\}, E) = \{S' \rightarrow E\}$   
 $\text{GOTO}(\{E \rightarrow .T-E\}, T) = \{E \rightarrow T-E, T \rightarrow .F \times T, T \rightarrow .F, F \rightarrow .id\}$   
 $\text{GOTO}(\{E \rightarrow .T\}, '-') = \{E \rightarrow T-E, T \rightarrow .F \times T, T \rightarrow .F, F \rightarrow .id\}$   
 $\text{GOTO}(\{T \rightarrow .F \times T\}, F) = \{T \rightarrow F \times T\}$   
 $\text{GOTO}(\{E \rightarrow T-E, T \rightarrow .F \times T, T \rightarrow .F, F \rightarrow .id\}, F) = \{T \rightarrow F \times T\}$   
 $\text{GOTO}(\{T \rightarrow F \times T\}, T) = \{T \rightarrow F \times T, F \rightarrow .id\}$   
 $\text{GOTO}(\{T \rightarrow F \times T, F \rightarrow .id\}, '-') = \{E \rightarrow T-E\}$

Computing the sets of LR(1) items by combining the LR(0) items and their GOTO sets.

**I0 = Closure({S' → .E})**  
**I1 = GOTO({S' → .E}, E)**  
**= {S' → E.}**  
**I2 = GOTO({E → .T-E, T → .F×T, T → .F, F → .id}, T)**  
**= {E → T-E, T → .F×T, T → .F, F → .id}**  
**I3 = GOTO({E → T-E, T → .F×T, T → .F, F → .id}, F)**  
**= {T → F×T}**  
**I4 = GOTO({T → F×T, F → .id}, T)**  
**= {T → F×.T, F → .id}**  
**I5 = GOTO({E → T-E., T → .F×T, T → .F, F → .id}, '-')**  
**= {E → T-E.}**

Here are the sets of items:

**I0:**  
**S' → .E**

**I1:**  
**S' → E.**

**I2:**  
**E → T-E**  
**E → .T**  
**T → .F×T**  
**T → .F**  
**F → .id**

**I3:**  
**T → F×T**

**I4:**  
**T → F×.T**  
**T → .F**  
**F → .id**

**I5:**  
**E → T-E.**  
**T → .F×T**  
**T → .F**  
**F → .id**

## LR(1) Parsing Table:

	id	x	-	\$	E	T
0	s2				1	
1				acc		
2	s2	s3	s5			4
3	r5		r5	r5		
4	r4	s3	r4	r4		
5	r3	r3	r3	r3		

Python file attached, and code uploaded on GitHub. Added you as a collaborator.

<https://github.com/maaz29/Compiler-Assignment-2>

Kindly accept the collaborator invitation. Thank you.