

Instructions to run the ParseIT Tool

1. Open the terminal and go to the location of ParseIT directory.
2. While running the jar file, you need to give a file containing a CFG (Context Free Grammar). Some sample grammar files are contained in the sub-directory named 'Grammars'.
3. Run jar file by the following command-

`java -jar ParseIT.jar path/file.txt`

(Here path is the path to a grammar file.)

Eg: If you want to give 'grammar15.txt' file (contained in Grammars sub-directory), then the command is -

`java -jar ParseIT.jar Grammars/grammar15.txt`

4. Now the selected grammar with a menu will be displayed on the screen.
5. Select one of the domain (for which problems will be generated) by typing the option number like 1 for first set, 2 for follow set and so on.
6. Suppose you select 1, then the tool generates a problem on first set of each of the non-terminals in the grammar. Select the all terminals which combinedly form the answer to the problem. For this, type the options space separately.

Eg:

Grammar: $S \rightarrow a A B b$, $A \rightarrow c \mid \epsilon$, $B \rightarrow d \mid \epsilon$

Problem: Which symbols should be included in $FIRST[A]$?

Options: d b c a epsilon

Answer: c epsilon

7. If the answer to the previous problem is incorrect, then the tool generates hint problems to direct you to the correct answer.

Eg: If you choose options d and epsilon in the previous example, then it will generate a series of hint problems as -

- a) According to which of the following rules, 'd' is a part of $FIRST[A]$?
 - 1) If X is a terminal, then $FIRST(X)$ is $\{X\}$.
 - 2) If X is nonterminal and $X \rightarrow \alpha$ is a production, then add a to $FIRST(X)$. If $X \rightarrow \epsilon$ is a production, then add epsilon to $FIRST(X)$.
 - 3) If $X \rightarrow Y_1 Y_2 \dots Y_k$ is a production, then for all i such that all of Y_1, \dots, Y_{i-1} are nonterminals and $FIRST(Y_j)$ contains epsilon for $j = 1, 2, \dots, i-1$ (i.e $Y_1 Y_2 \dots Y_{i-1} \rightarrow \epsilon$), add every non-epsilon symbol in $FIRST(Y_i)$ to $FIRST(X)$. If epsilon is in $FIRST(Y_j)$ for all $j = 1, 2, \dots, k$, then add epsilon to $FIRST(X)$.
 - 4) No valid rule for this symbol.
- b) Should 'c' be included in $FIRST[A]$?
- c) According to which of the following rules, 'c' is a part of $FIRST[A]$?
 - 1) If X is a terminal, then $FIRST(X)$ is $\{X\}$.
 - 2) If X is nonterminal and $X \rightarrow \alpha$ is a production, then add a to $FIRST(X)$. If $X \rightarrow \epsilon$ is a production, then add epsilon to $FIRST(X)$.

3) If $X \rightarrow Y_1 Y_2 \dots Y_k$ is a production, then for all i such that all of Y_1, \dots, Y_{i-1} are nonterminals and $\text{FIRST}(Y_i)$ contains epsilon for $j = 1, 2, \dots, i-1$ (i.e. $Y_1 Y_2 \dots Y_{i-1} \rightarrow \epsilon$), add every non-epsilon symbol in $\text{FIRST}(Y_i)$ to $\text{FIRST}(X)$. If epsilon is in $\text{FIRST}(Y_j)$ for all $j = 1, 2, \dots, k$, then add epsilon to $\text{FIRST}(X)$.

4) No valid rule for this symbol.

Type the correct answer. If an answer to a hint problem is incorrect, then the tool repeats the same hint problem.

8. The scenario described above remains the same for every domain of problems.
9. For parsing tables, there are two levels which differ only in the types of hint problems. In first level regular hint problems are generated. In the second level, hint problems containing parsing on input strings are generated.

Eg: The parsing table for the used in above examples:

	c	a	b	d	\$
=====	=====	=====	=====	=====	=====
A	A->c				
S		S->a A B b			
B					

Problem: Which grammar rule should be included in the cell [B, d] of the parsing table?

Options: A -> c A -> epsilon S -> a A B b B -> d B -> epsilon ERROR

Correct Answer: B -> d

User Answer: B -> epsilon

Hint Problem:

Stack	Input	Output
=====	=====	=====
\$S	a d b \$	
\$bBAa	a d b \$	S -> a A B b
\$bBA	d b \$	
\$bB	d b \$	A -> epsilon
\$b	d b \$	B -> epsilon

LL Parsing on this input string is not working correctly due to the wrong entry made in the cell [B, d]. Can you fix the error by selecting the correct choice?

A -> c A -> epsilon S -> a A B b B -> d B -> epsilon ERROR

Type-in the correct answer (B -> d in this case) to move to the next question. Type the rules space separately.

10. For parsing moves, type-in a input string(space separately) you would like to parse. Like for the grammar in above example, one input string can be: acb. So type as: a c b

11. The tool then displays the problem as:

Stack	Input	Output
=====		
\$S	a c b \$	
\$bBAa	a c b \$	S -> a A B b
\$bBA	c b \$	

What will be the next move?

Type-in the correct answer double space separately. Like the next move of above parsing should be typed as:

\$ b B c c b \$ A -> c

If the answer to the problem is incorrect, the tool generates the regular hint problems described earlier.

12. If you want to exit in between, then press Ctrl+c.