

CYCLOPS

About

Cyclops is a software system for teaching LL(1) parsing. Cyclops provides an interactive environment to learn LL(1) parsing. It uses formal method techniques (i.e., SMT solvers) to give feedback to the user in case of wrong entries of the LL(1) parse table. Cyclops is built on the published work Parse Condition:

Parsing [Symbolic Encoding of LL\(1\)](#)

Publications

Dhruv Singal, Palak Agarwal, Saket Jhunjhunwala and Subhajit Roy. Parse Condition: Symbolic Encoding of LL(1) Parsing. Parsing

[In LPAR-22. 22nd International Conference on Logic for Programming, Artificial Intelligence and Reasoning, Awassa, Ethiopia, 16-21 November. 2018.](#)

Register Here

Email

rasingham64@gmail.com

Password

.....

SIGN UP

Already have an account? Sign In

email already
taken or details
wrong

Register Here

Email

Password

SIGN UP

Already have an account? Sign In

localhost:3000 says

Proceed to login

OK

Cancel

Register Here

Email

Password

SIGN UP

Already have an account? Sign In

Login

Email

Password

Sign In

Not having an account? Register Now

CYCLOPS

About

Cyclops is a software system for teaching LL(1) parsing. Cyclops provides an interactive environment to learn LL(1) parsing. It uses formal method techniques (i.e., SMT solvers) to give feedback to the user in case of wrong entries of the LL(1) parse table. Cyclops is built on the published work Parse Condition:

Parsing [Symbolic Encoding of LL\(1\)](#)

Publications

Dhruv Singal, Palak Agarwal, Saket Jhunjhunwala and Subhajit Roy. Parse Condition: Symbolic Encoding of LL(1) Parsing. Parsing

[In LPAR-22. 22nd International Conference on Logic for Programming, Artificial Intelligence and Reasoning, Awassa, Ethiopia, 16-21 November. 2018.](#)

Problem statement:

Your group was asked to create an LL(1) parser for a given grammar. As you were busy, your group members finished the assignment and sent the solution to you for verification. Clever you---you decided to use [LL\(1\) parser generator](#) to check the solution. But, to your dismay the constructed parse table is wrong!

Now, you know that your group members are careful, they could have created **exactly one error** while following the steps to LL(1) parser creation. To be more precise, they made a mistake in exactly one constraint of [first set](#), [follow set](#) and [parse table](#) construction---at that too on a single non-terminal/terminal/production. You are required to locate the bug.

For your assistance, we have provided a bot, [Cyclops](#)¹, that will help you debug the fault. Given a grammar and an incorrect parse table, it ranks the potential error and you may use it for directing your debugging effort. Like you, Cyclops does not have any idea of the error and hence, the actual fault may not appear at the top of the ranked list, or may not appear at all.

Your TA would have provided a passcode to access the site (and identify your group). You should attempt the Grammar/Tasks assigned to you (we have kept all tasks open, in case you want to explore other tasks too); however, only the tasks assigned to you will be graded.

We provide a possible sample solution that we expect from you: **The error was in the second constraint of First Set construction corresponding to the non-terminal 'A'. The terminal 'b' was present as the first symbol in the body of production $A \rightarrow b A c$, and hence, should have been included in the first-set of 'A'. Adding 'b' to FIRST(A), adds $A \rightarrow b A c$ to the parse table corresponding to non-terminal 'A' and lookahead 'b'. This is the correct parse table.**

You must provide an honest feedback about if [Cyclops](#) helped you solve the problem and how (about 50 words). We will grade you on the quality of this feedback. Please be honest; both positive and negative feedback is equally important to us.

Important: We intend to use your (anonymized) feedback in future publications. We assume that you provide your consent for the same when you provide feedback. In case you are not comfortable with the same, please send a mail to subhajit@iitk.ac.in and pkalita@cse.iitk.ac.in.

Constraints for parse table creation:

First set constraints:

1. If X is a terminal symbol then $FIRST(X) = X$.
2. If X is a non terminal symbol and $X \rightarrow Y_1, Y_2, \dots Y_k$ is a production, then
If for some i , a is in $FIRST(Y_i)$ and ϵ is in all of $FIRST(Y_j)$ (such that $j < i$) then
 a is in $FIRST(X)$.
If ϵ is in $FIRST(Y_1) \dots FIRST(Y_k)$ then ϵ is in $FIRST(X)$.
3. If $X \rightarrow \epsilon$ is a production then ϵ is in $FIRST(X)$.

Follow set constraints:

1. Place $\$$ in $FOLLOW(S)$, where S is the start symbol and $\$$ is the input right endmarker.
2. If there is a production $A \rightarrow \alpha B \beta$, then everything in $FIRST(\beta)$ except ϵ is in $FOLLOW(B)$.
3. If there is a production $A \rightarrow \alpha B$, or a production $A \rightarrow \alpha B \beta$, where $FIRST(\beta)$ contains ϵ , then everything in $FOLLOW(A)$ is in $FOLLOW(B)$.

3. If $X \rightarrow \epsilon$ is a production then ϵ is in $FIRST(X)$.

Follow set constraints:

1. Place $\$$ in $FOLLOW(S)$, where S is the start symbol and $\$$ is the input right endmarker.
2. If there is a production $A \rightarrow \alpha B \beta$, then everything in $FIRST(\beta)$ except ϵ is in $FOLLOW(B)$.
3. If there is a production $A \rightarrow \alpha B$, or a production $A \rightarrow \alpha B \beta$, where $FIRST(\beta)$ contains ϵ , then everything in $FOLLOW(A)$ is in $FOLLOW(B)$.

Parse Table set constraints:

1. For each terminal a in $FIRST(\alpha)$, add $A \rightarrow \alpha$ to $M[A, a]$.
2. If ϵ is in $FIRST(\alpha)$, then for each terminal b in $FOLLOW(A)$, add $A \rightarrow \alpha$ to $M[A, b]$. If ϵ is in $FIRST(\alpha)$ and $\$$ is in $FOLLOW(A)$ add $A \rightarrow \alpha$ to $M[A, \$]$ as well.

Team



Pankaj Kumar Kalita
PhD student



Sumit Lahiri
PhD student



Dr. Subhajit Roy
Associate Professor

React App

cyclops - Authentication with Ide

console.firebase.google.com/u/0/project/cyclops-5acac/authentication/users

Go to docs

Firebase

Project Overview

Project shortcuts

Authentication

Analytics Dashboard

What's new

Extensions

Product categories

Build

Release & Monitor

Analytics

Engage

All products

Customize your nav!

Spark

No-cost \$0/month

Upgrade

cyclops

Authentication

with Identity Platform

Users

Sign-in method

Templates

Usage

Settings

Search by email address, phone number, or user UID

Add user

Identifier	Providers	Created	Signed In	User UID
codechef.iitk@gmail.com		Oct 17, 2022	Oct 17, 2022	GSNEMLYPPUf5jm3OrCiBtCZaE393
rasingham64@gmail.com		Oct 17, 2022	Oct 17, 2022	xHm3gWHuoTV6aNSAEH5aU2
shashvatsingham2001@g...		Oct 17, 2022	Oct 17, 2022	vGgB4A6Je9cj3QBDdFqohiE14
singham64@gmail.com		Sep 6, 2022	Oct 17, 2022	knmE2dbklsdwa4lunXYTf6PdR

Reset password

Disable account

Delete account

Configure MFA

Rows per page: 50 1 - 4 of 4

Fun with LL(1)

[Home](#) [Solve](#) [View](#) [About](#)

[Log In](#)

For any technical difficulties or bugs regarding this site please send a mail to pkalita@cse.iitk.ac.in and subhajit@iitk.ac.in.

Problem statement:

Your group was asked to create an LL(1) parser for a given grammar. As you were busy, your group members finished the assignment and sent the solution to you for verification. Clever you---you decided to use **LL(1) parser generator** to check the solution. But, to your dismay the constructed parse table is wrong!

Now, you know that your group members are careful, they could have created **exactly one error** while following the steps to LL(1) parser creation. To be more precise, they made a mistake in exactly one constraint of **first set**, **follow set** and **parse table** construction---at that too on a single non-terminal/terminal/production. You are required to locate the bug.

For your assistance, we have provided a bot, **Cyclops**, that will help you debug the fault. Given a grammar and an incorrect parse table, it ranks the potential error and you may use it for directing your debugging effort. Like you, Cyclops does not have any idea of the error and hence, the actual fault may not appear at the top of the ranked list, or may not appear at all.

Grammar

Please input LL(1) grammar

Generate Parse Table

First Set

If X is a Terminal ▼

Then $FIRST(X) = \{X\}$

If X is a Nonterminal ▼

If $X \rightarrow \epsilon$ is a Production ▼

Follow Set

Place \$ in $FOLLOW(S)$ ▼

If there is a production $A \rightarrow \alpha B \beta$ ▼

If there is a production $A \rightarrow \alpha B$ or $A \rightarrow \alpha B \beta$ ▼

Parse Table Constraints

For each terminal a in $FIRST(\alpha)$ ▼

If ϵ is in $FIRST(\alpha)$ ▼

Grammar

```
E -> T X
T -> ( E
T -> int Y
X -> + E
X -> eps
Y -> * T
Y -> eps
```

Generate Parse Table

Number Rules

1	E -> T X
2	T -> (E
3	T -> int Y
4	X -> + E
5	X ->

) int +

	0	0	0
	0	0	0
2		0	0
7		0	0

Submit

First Set

If X is a Terminal ▼

If X is a Nonterminal ▼

If $X \rightarrow \epsilon$ is a Production ▼

Follow Set

Place \$ in $FOLLOW(S)$ ▼

If there is a production $A \rightarrow \alpha B \beta$ ▼

If there is a production $A \rightarrow \alpha B$ or $A \rightarrow \alpha B \beta$ ▼

Parse Table Constraints

For each terminal a in $FIRST(\alpha)$ ▼

If ϵ is in $FIRST(\alpha)$ ▼

Grammar

Please input LL(1) grammar

Generate Parse Table

Instructions

Enter semicolon separated values in the specified areas of terminals, non-terminals and grammar.

The format of grammar should be $S \rightarrow S(S)$ i.e LHS and RHS should be separated by a ' \rightarrow ' block only, otherwise you would get a syntax error. All the blank spaces would be neglected.

Then press generate parse table button for creation of parse table.

Fill the parse table and press submit button.

Close

First Set

is a Terminal ▼

 $FIRST(X) = \{X\}$

is a Nonterminal ▼

 $\rightarrow \epsilon$ is a Production ▼

Follow Set

Place \$ in $FOLLOW(S)$ ▼If there is a production $A \rightarrow \alpha B \beta$ ▼If there is a production $A \rightarrow \alpha B$ or $A \rightarrow \alpha B \beta$ ▼

Parse Table Constraints

For each terminal a in $FIRST(\alpha)$ ▼If ϵ is in $FIRST(\alpha)$ ▼