

Evaluation Item	Unacceptable (0)	Acceptable (1)		Comments
<b>Verifiability</b> Note: If this item is “unacceptable”, the rest of the assignment will not be marked.	The submission lacks a script for validating the parser, or the script provided could not be used to verify the parser.	The submission includes (a) a set of tests for verifying the parser and (b) a script to build your parser and to parse those tests. That script can be used on a lab machine to verify the parser outputs.		<ul style="list-style-type: none"><li>A set of test cases can be easily run using pytest.</li></ul> <div>1/1</div>
Evaluation Item	Unacceptable (0)	Marginal (1-2)	Proficient (3)	Comments
<b>Correctness</b>	The backend does not generate legal code in the desired output language.	The backend generates legal code in the desired output language (with small exceptions). Invalid code may be generated if the input is erroneous.	The backend generates legal code in the desired output language. When it encounters an error in the input, an error is reported.	<ul style="list-style-type: none"><li>Everything described from the language spec seems to be incorporated into the compiler.</li><li>There should also be test cases to check for illegitimate input (to see if the typechecker works properly).</li></ul> <div>3/3</div>

<b>Completeness</b>	The backend fails to generate code for significant portions of the input language.	The backend fails to generate code for one or two legal structures.	The backend generates legal code for all structures identified in the language spec.	<ul style="list-style-type: none"> <li>The backend seems to generate legal code for all structures identified.</li> </ul>
				3/3
<b>Evaluation Item</b>	<b>Unacceptable (0)</b>	<b>Marginal (1-2)</b>	<b>Proficient (3)</b>	<b>Comments</b>
<b>Documentation</b>	The submission document is missing one of the required elements.	The submission document contains all three required elements but does not <i>clearly</i> describe the IR or process for verifying the parser.	The submission contains a document that (a) describes the IR clearly and explains how it is transformed, (b) lists language structures not parsed, and (c) explains how to use the provided script and tests.	<ul style="list-style-type: none"> <li>It was a bit hard to find where the three required elements were located in the documentation, and things could be better organized. For instance, there is no clear list for language structures not parsed or whether all structures were parsed.</li> <li>The type checker documentation is in a separate PDF and that could be combined.</li> <li>Consider including a table of contents and clear headings + subheadings for easier navigation.</li> </ul>
				2/3

**Additional Notes:**

- None.

**Group Members:**

1004545842	CHEN	LITAO	litao.chen@mail.utoronto.ca	chenlita
1003403872	SELLATHURAI	JATHAVAN	jathavan.sellathurai@mail.utoronto.ca	sellat16
1004910670	YIN	YIFEI	yifei.yin@mail.utoronto.ca	yinyife2

**Score:** 9 /10