Marker:

Programming language feature is implemented and tested in the assignment	
Y/N	**Feature**
0	class declarations
0	data member declarations
0	member function declarations
0	inheritance list
0	private/public members
0	free function definitions
0	member function definitions
0	variable declaration block
0	int, float, variable declarations
0	array variable declarations
0	if statement
0	while statement
0	read/write statement
0	return statement
0	assignment statement
0	complex street structures involving () and .s, and expressions as array indexes
0	complex expressions involving add,op, multop, relop, unaryop, street, function calls

Notes	Weight %	ratio per mark	letter			
	0%	0.00%		1	**Document Section 1 - Presentation of 1(1) grammar, and realted transformations**	1 :
	0%	0.00%	X	1.1	Clarity - presentation of grammar and the problems/transformations	1.1 : X :
	0%	0.00%	X	1.2	Completeness - all problems with the grammar are listed/explained	1.2 : X :
	0%	0.00%	X	1.3	Statement of changed specification or/aus original specifications in the assignment statement (if applicable)	1.3 : X :
	0%	0.00%		2	**Document Section 2 - FIRSTP/Ssc0W sets**	2 :
	0%	0.00%	X	2.1	Clarity - presentation of FIRSTP/Ssc0W sets is clear and understandable	2.1 : X :
	0%	0.00%	X	2.2	Completeness - FIRSTP/Ssc0W sets are given for all non-terminal symbols	2.2 : X :
	0%	0.00%		3	**Document Section 3 - Description/rationale of the overall structure of the solution and the roles of the individual components used in the applied solution to the stated problem**	3 :
	0%	0.00%	X	3.1	Clarity - presentation of the design description is clear and understandable	3.1 : X :
	0%	0.00%	X	3.2	Completeness - all implementing components are mentioned in the design description	3.2 : X :
	0%	0.00%		4	**Document Section 4 - Description of tools/libraries/techniques used in the analysis/implementation. Description of other tools that might have been used. Justification of why the chosen tools were selected**	4 :
	0%	0.00%	X	4.1	All tools used in the analysis/implementation are mentioned.	4.1 : X :
	0%	0.00%	X	4.2	Justification for all tools.	4.2 : X :
	0%	30.00%		5	**Correct implementation of a top-down predictive parser according to the original grammar**	5 :
	0%	0.00%	X	5.1	class declarations	5.1 : X :
	0%	0.00%	X	5.2	data member declarations	5.2 : X :
	0%	0.00%	X	5.3	member function declarations	5.3 : X :
	0%	0.00%	X	5.4	inheritance list	5.4 : X :
	0%	0.00%	X	5.5	private/public members	5.5 : X :
	0%	0.00%	X	5.6	free function definitions	5.6 : X :
	0%	0.00%	X	5.7	member function definitions	5.7 : X :
	0%	0.00%	X	5.8	variable declaration block	5.8 : X :
	0%	0.00%	X	5.9	int, float, variable declarations	5.9 : X :
	0%	0.00%	X	5.10	array variable declarations	5.10 : X :
	0%	0.00%	X	5.11	if statement	5.11 : X :
	0%	0.00%	X	5.12	while statement	5.12 : X :
	0%	0.00%	X	5.13	read/write statement	5.13 : X :
	0%	0.00%	X	5.14	return statement	5.14 : X :
	0%	0.00%	X	5.15	assignment statement	5.15 : X :
	0%	0.00%	X	5.16	complex street structures involving () and .s, and expressions as array indexes	5.16 : X :
	0%	0.00%	X	5.17	complex expressions involving add,op, multop, relop, unaryop, street, function calls	5.17 : X :
	0%	0.00%		6	**Output - outputsource.txt file: Output of clear error messages (error description and location) is a file**	6 :
	0%	0.00%	X	6.1	Clarity - error messages include a line number where the error was found	6.1 : X :
	0%	0.00%	X	6.2	Clarity - error messages describe what specific kind of error was found	6.2 : X :
	0%	0.00%	X	6.3	Clarity - error messages include token/lexeme/lexeme that correspond to the error found	6.3 : X :
	0%	0.00%	X	6.4	Correctness - errors are correctly identified and correct syntax is not reported as errors	6.4 : X :
	0%	0.00%	X	6.5	Completeness - all errors are reported, no errors are missed in the assignment	6.5 : X :
	0%	10.00%		7	**Output - outderivation.txt: Output of a derivation corresponding to the parsing**	7 :
	0%	0.00%	X	7.1	Clarity - derivation is somehow readable	7.1 : X :
	0%	0.00%	X	7.2	Correctness - output is really a derivation in the strict sense of the term	7.2 : X :
	0%	0.00%		8	**Error recovery – the parser can recover from syntax errors**	8 :
	0%	0.00%	X	8.1	Error recovery is implemented - the parser can recover after an error	8.1 : X :
	0%	20.00%		9	**Test cases - completeness of testing (in addition to the provided files)**	9 :
	0%	0.00%	X	9.1	class declarations	9.1 : X :
	0%	0.00%	X	9.2	data member declarations	9.2 : X :
	0%	0.00%	X	9.3	member function declarations	9.3 : X :
	0%	0.00%	X	9.4	inheritance list	9.4 : X :
	0%	0.00%	X	9.5	private/public members	9.5 : X :
	0%	0.00%	X	9.6	free function definitions	9.6 : X :
	0%	0.00%	X	9.7	member function definitions	9.7 : X :
	0%	0.00%	X	9.8	variable declaration block	9.8 : X :
	0%	0.00%	X	9.9	int, float, variable declarations	9.9 : X :
	0%	0.00%	X	9.10	array variable declarations	9.10 : X :
	0%	0.00%	X	9.11	if statement	9.11 : X :
	0%	0.00%	X	9.12	while statement	9.12 : X :
	0%	0.00%	X	9.13	read/write statement	9.13 : X :
	0%	0.00%	X	9.14	return statement	9.14 : X :
	0%	0.00%	X	9.15	assignment statement	9.15 : X :
	0%	0.00%	X	9.16	complex street structures involving () and .s, and expressions as array indexes	9.16 : X :
	0%	0.00%	X	9.17	complex expressions involving add,op, multop, relop, unaryop, street, function calls	9.17 : X :
	0%	0.00%		10	**Reasonable/correct use of tools/libraries/techniques in the analysis/implementation.**	10 :
	0%	0.00%	X	10.1	The program never crashes or throws exceptions	10.1 : X :
	0%	0.00%	X	10.2	Tools presented in the labs are used appropriately, or comparable tools are used appropriately	10.2 : X :

Marker instructions

Enter values only in the red cells. Everything else is calculated automatically.
In Column G, enter either A, B, C, or F for each marking element
Enter notes in column A. These notes will be then automatically uploaded on moodle as feedback to the students.

Designer instructions

Add/remove some lines with red cells to add/remove evaluation criteria
When you add lines, add them only in the middle of the area, or else the calculation is going to be wrong
To add lines, add a new row, then copy into it one of the existing rows.

Grading rubric	ratio		base ratio	Total marks
Document Section 1 - Presentation of 1(1) grammar, and realted transformations	0	0.00	0.00	0.00
Document Section 2 - FIRSTP/Ssc0W sets	0	0.00	0.00	0.00
Document Section 3 - Description/rationale of the overall structure of the solution and the roles of t	0	0.00	0.00	0.00
Document Section 4 - Description of tools/libraries/techniques used in the analysis/implementation	0	0.00	0.00	0.00
Correct implementation of a top-down predictive parser according to the original grammar	1.5	0.05	0.00	0.00
Output - outputsource.txt file: Output of clear error messages (error description and location) is a f	0	0.00	0.00	0.00
Output - outderivation.txt: Output of a derivation corresponding to the parsing	0	0.00	0.00	0.00
Error recovery – the parser can recover from syntax errors	0	0.00	0.00	0.00
Test cases - completeness of testing (in addition to the provided files)	0.8	0.00	0.00	0.00
Reasonable/correct use of tools/libraries/techniques in the analysis/implementation	0.5	0.00	0.00	0.00
Total	40	0.00		0.00