

Grammar Smells

Misformat <pre>a: b; c: d;</pre>	Underuse <pre>args: arg+ stmts: stmt stmt stmts</pre>	Priorities <pre>xor: and xor "^" and; and: cmp and "&" cmp;</pre>	Combo <pre>a : b? b : c?</pre>	Deprecated <pre>a : [b];</pre>	Almost <pre>D: [1-9]+; L: [b-zA-Z]</pre>	Uncluster <pre>a: "A" b; c: "C" d; d: "D" c; b: "B" a;</pre>
Misnomer <pre>stmt ::= IF condition _THEN b;</pre>	Overspec <pre>a: ! "b" & "c";</pre>	Singleton <pre>foo: one of bar</pre>	Chant <pre>D : [0-9]; L : any letter</pre>	Exotic <pre>syntax A = "a" > "b";</pre>	Confusing Entry <pre>an: "a"+ bn bn: "b"+ cn cn: "c"+ an</pre>	Unsequence <pre>a: b c; c: d e; d: f e; b: e f;</pre>
SayMyName <pre>statement seperate arguement grammer</pre>	Factoring <pre>if: IF c TH s IF c TH s EL s</pre>	Shotgun <pre>args: ![]]+</pre>	Echo <pre>foo: bar; bar: b a r; foo: bar;</pre>	Common <pre>a: b c d; e: b c f; g: b c h;</pre>	Dead <pre>foo: bar; z: foo+;</pre>	MiddleStart <pre>an: "a"+; bn: "b"+; S:an bn cn; cn: "c"+;</pre>
ZigZag <pre>a ::= b c; a ::= d;</pre>	Recursion <pre>exp: exp op exp</pre>	NoDefault <pre>stmt: map /case /if;</pre>	Clone <pre>case: "CASE" cof: c "OF" c: "CASE"</pre>	Permuted <pre>foo: a b; bar: b a;</pre>	Bottom <pre>Int ::= D+;</pre>	BadLayout <pre>L:[\n\t\]+ "/" .* \$</pre>
Splat <pre>a ::= b c; x ::= y z; a ::= d e;</pre>	Superset <pre>Id: [a-z]+</pre>	Action <pre>a: b {c()};</pre>	Foldable <pre>case: "CASE" cof: "CASE" "OF"</pre>	Lookalike <pre>if: "IF" ifB "END" do: "DO" doB "END"</pre>	Debt <pre>Expr := ε; // TODO</pre>	Preprocessor <pre>pragma: "#" pword ppcontent "#end";</pre>
Ouroboros <pre>a : b c; b : d e; d : f g; f : a h;</pre>	Soulmates <pre>co-changed</pre>	Spillover <pre>block: (stmt ";")+</pre>	Mythic <pre>never used</pre>	Chain <pre>condstmt: ifstmt; ifstmt: "IF" ... ;</pre>	Throwaway <pre>used once</pre>	Weak <pre>if: ifstart ifmid ifend; ifstart: "IF" "("; ifmid: cond ")" "{";</pre>
Rivalry <pre>name : id stdname;</pre>	2Modular <pre>too many modules</pre>	2Ramosse <pre>high cyclomatic complexity</pre>	2Recursive <pre>too many recursive nonterminals</pre>	2Nested <pre>too many subgroups</pre>	2Coupled <pre>high coupling low cohesion</pre>	Ghost <pre>f: Id("."Id)+ "(" args ")";</pre>
Diamond <pre>t : l r; l : b; r : b;</pre>	2Tall <pre>some nonterminals are too far from root</pre>	Lonely <pre>should be split into several</pre>	Greedy <pre>some modules do too much</pre>	Lazy <pre>some modules do too little</pre>	2Wide <pre>rhs too long</pre>	Multitool <pre>decl: name ":" name;</pre>

Interactive at <http://slebok.github.io/grass>

Mats Stijlaart & Vadim Zaytsev

raincode LABS
 compiler experts

