**Slovenská technická univerzita v Bratislave**

**Fakulta informatiky a informačných technológií**

**Daniel Valocký, Richard Pastorek**

**Projekt č.3 SMALL**

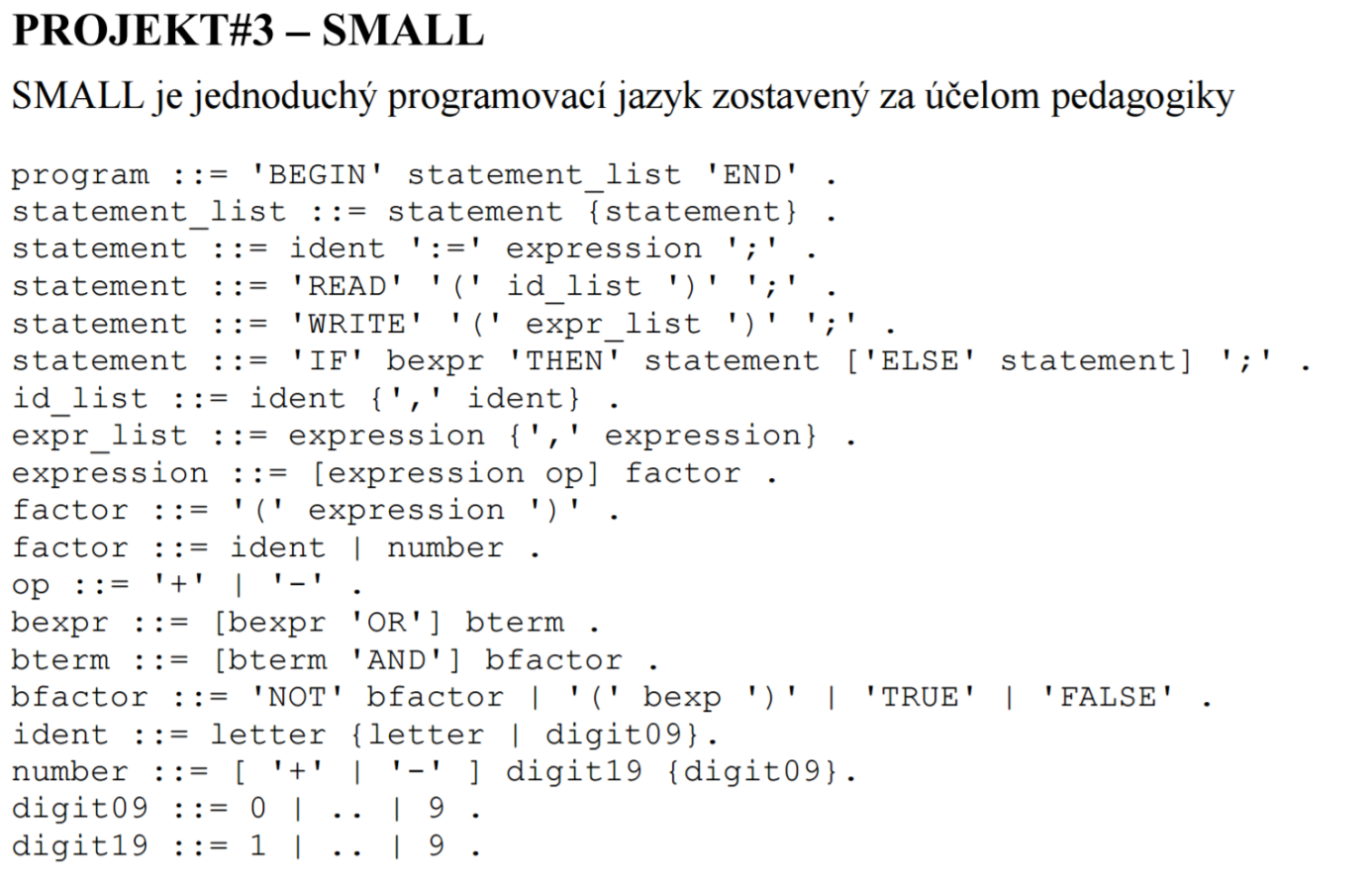
Semestrálna práca

Študijný program: Softvérové inžinierstvo

Cvičenie:  Pondelok 15:00

December, 2016

# Zadanie



# Analýza a návrh

## Príklad odvodenia vety daného jazyka

1. program
2. BEGIN statement\_list END
3. BEGIN statement END
4. BEGIN IF bexpr THEN statement ELSE statement ; END
5. BEGIN IF bterm THEN statement ELSE statement ; END
6. BEGIN IF bfactor THEN statement ELSE statement ; END
7. BEGIN IF NOT bfactor THEN statement ELSE statement ; END
8. BEGIN IF NOT TRUE THEN statement ELSE statement ; END
9. BEGIN IF NOT TRUE THEN WRITE ( expr\_list ) ; ELSE statement ; END
10. BEGIN IF NOT TRUE THEN WRITE ( expression ) ; ELSE statement ; END
11. BEGIN IF NOT TRUE THEN WRITE ( expression op factor ) ; ELSE statement ; END
12. BEGIN IF NOT TRUE THEN WRITE ( factor op factor ) ; ELSE statement ; END
13. BEGIN IF NOT TRUE THEN WRITE ( number op factor ) ; ELSE statement ; END
14. BEGIN IF NOT TRUE THEN WRITE ( 3 op factor ) ; ELSE statement ; END
15. BEGIN IF NOT TRUE THEN WRITE ( 3 - factor ) ; ELSE statement ; END
16. BEGIN IF NOT TRUE THEN WRITE ( 3 - number ) ; ELSE statement ; END
17. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE statement ; END
18. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( expr\_list ) ; ; END
19. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( expression ) ; ; END
20. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( expression op factor ) ; ; END
21. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( factor op factor ) ; ; END
22. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( number op factor ) ; ; END
23. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( 3 op factor ) ; ; END
24. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( 3 + factor ) ; ; END
25. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( 3 + number ) ; ; END
26. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( 3 + 2 ) ; ; END

Výsledok: BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( 3 + 2 ) ; ; END

## Príklady viet daného jazyka

### Akceptované (správne) vety

1. BEGIN var := ( 15 ) ; END
2. BEGIN IF NOT TRUE THEN WRITE ( 3 - 2 ) ; ELSE WRITE ( 3 + 2 ) ; ; END
3. BEGIN READ ( file , file2 ) ; WRITE ( 1 ) ; IF TRUE THEN WRITE ( file ) ; ELSE WRITE ( file2 ) ; ; file := - 1 ; file2 := file ; END
4. BEGIN var := false ; IF FALSE OR TRUE AND ( FALSE OR TRUE ) THEN var := true ; ; WRITE ( value , after , if , is , var ) ; END
5. BEGIN var2 := var5 + ( 9685 - 789 ) ; IF NOT TRUE THEN WRITE ( num1 + num2 ) ; ELSE IF FALSE THEN var3 := ahoj ; ELSE var3 := ahoj2 ; ; ; END
6. BEGIN WRITE ( num8 - 5 ) ; END
7. BEGIN var2 := var7 ; IF FALSE THEN var9 := va6 ; ELSE IF NOT FALSE THEN var4 := var5 + ( 9685 - 789 ) ; ; ; IF NOT TRUE THEN WRITE ( num1 + num2 ) ; ELSE IF FALSE THEN var3 := ahoj ; ELSE var3 := ahoj2 ; ; ; var10 := var8 + ( var6 - 789 ) ; END
8. BEGIN READ ( var11 , var158 ) ; var95 := ( - 2131 ) ; IF TRUE OR TRUE AND NOT NOT NOT NOT FALSE THEN WRITE ( ( + 2011 ) ) ; ; var143 := 2756 ; IF FALSE OR NOT FALSE THEN var64 := 2943 ; ELSE var229 := + 2800 ; ; WRITE ( + 3383 ) ; END
9. BEGIN var438 := + 1814 ; END
10. BEGIN IF FALSE THEN WRITE ( ( ( - 3725 ) ) - var114 ) ; ELSE IF ( NOT ( ( ( NOT TRUE ) ) OR ( TRUE ) ) ) AND TRUE AND NOT FALSE OR NOT TRUE THEN IF ( FALSE AND NOT FALSE OR NOT FALSE ) THEN READ ( var150 , var298 , var281 ) ; ; ; ; var473 := 1387 + 4886 + var349 - 4308 ; IF NOT NOT FALSE AND NOT ( ( FALSE ) AND ( FALSE AND NOT TRUE ) ) THEN READ ( var327 ) ; ELSE READ ( var84 ) ; ; END
11. BEGIN IF FALSE THEN var495 := ( var94 ) + ( - 4414 - + 2785 ) + 362 ; ELSE READ ( var171 , var313 ) ; ; WRITE ( - 2023 ) ; END
12. BEGIN IF TRUE THEN var486 := + 4371 - - 762 ; ELSE WRITE ( + 2091 , - 465 , + 2766 , 209 , + 2725 + var317 - + 880 ) ; ; END
13. BEGIN IF ( TRUE AND FALSE AND FALSE OR TRUE ) AND NOT FALSE THEN IF FALSE OR NOT TRUE OR FALSE OR NOT FALSE THEN WRITE ( + 1586 , - 3069 , - 606 , var440 , ( - 2151 - + 2533 - + 2677 ) , + 4611 - + 4308 ) ; ; ; IF TRUE AND FALSE AND FALSE OR FALSE THEN READ ( var479 , var192 ) ; ELSE IF FALSE THEN READ ( var339 , var113 ) ; ; ; WRITE ( 3408 ) ; READ ( var284 ) ; READ ( var124 , var324 , var104 , var315 ) ; IF NOT ( FALSE ) THEN var259 := - 1440 ; ELSE READ ( var211 , var447 ) ; ; READ ( var97 ) ; READ ( var295 ) ; var195 := ( + 3597 - ( var348 ) + + 2298 - - 1662 + + 866 - - 3741 - - 1001 ) - ( var272 ) + - 3823 ; END
14. BEGIN WRITE ( ( ( ( - 1823 ) - ( ( + 3451 ) ) ) ) ) ; var380 := var463 ; END
15. BEGIN IF TRUE THEN IF TRUE THEN READ ( var405 ) ; ELSE READ ( var468 , var72 , var96 ) ; ; ELSE READ ( var173 , var133 ) ; ; READ ( var76 ) ; END
16. BEGIN WRITE ( + 2598 , var368 , - 4826 - 3576 ) ; READ ( var394 ) ; END
17. BEGIN var411 := + 4267 ; IF NOT FALSE AND TRUE THEN READ ( var349 ) ; ELSE WRITE ( var301 ) ; ; END

### Neakceptované (nesprávne) vety

## Prepis

Toto nie je finálny prepis nakoľko sme sa neskôr rozhodli, že správnosť využitia pravidiel týkajúcich sa neterminálov "D" a "L" overíme už v lexikálnom analyzátore. Finálny prepis sa nachádza v kapitole 2.5.

Tabuľka 1 Prepis terminálov a neterminálov na nami zadefinované znaky

|  |  |  |  |
| --- | --- | --- | --- |
| Terminály | Znak | Neterminály | Znak |
| 'BEGIN' | a | program | S |
| 'END' | b | statement\_list | B |
| ':=' | c | statement | C |
| ';' | d | id\_list | F |
| 'READ' | e | expr\_list | G |
| '(' | f | expression | E |
| ')' | g | factor | K |
| 'WRITE' | h | op | J |
| 'IF' | i | bexpr | H |
| 'THEN' | j | bterm | M |
| 'ELSE' | k | bfactor | N |
| ',' | l | ident | D |
| '+' | m | number | L |
| '-' | o | digit09 | Q |
| 'OR' | p | digit19 | R |
| 'NOT' | r | letter | P |
| 'TRUE' | s |  |  |
| 'FALSE' | t |  |  |
| 'AND' | q |  |  |
| '0' | u |  |  |
| 1' | x |  |  |
| '9' | y |  |  |
| 'z' | z |  |  |

## First a Follow

Tabuľka 2 Pravidlá, čísla paravidiel a vymenovanie first a follow

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Neterminály | Pravidlá | Čísla pravidiel | First | Follow |
| S | aBb | 1 | {a} | {$} |
| B | CX | 2 | {z,e,h,i} | {b} |
| X | CX|ε | 3|4 | {z,e,h,i,ε} | {b} |
| C | DcEd|efFgd|hfGgd|iHjCYd | 5|6|7|10 | {z,e,h,i} | {z,e,h,i,b,k,d} |
| Y | kC|ε | 8|9 | {k,ε} | {d} |
| F | DZ | 11 | {z} | {g} |
| Z | lDZ|ε | 12|13 | {l,ε} | {g} |
| G | EW | 14 | {f,z,m,o,x,y} | {g} |
| W | lEW|ε | 15|16 | {l,ε} | {g} |
| E | KE'' | 17 | {f,z,m,o,x,y} | {d,l,g} |
| E' | JKE'' | 51 | {m,o} | {d,l,g} |
| E'' | E'|ε | 18|19 | {m,o,ε} | {d,l,g} |
| K | fEg|D|L | 20|21|22 | {f,z,m,o,x,y} | {m,o,d,l,g} |
| J | m|o | 23|24 | {m,o} | {f,z,m,o,x,y} |
| H | MH'' | 25 | {r,f,s,t} | {j,g} |
| H' | pMH'' | 52 | {p} | {j,g} |
| H'' | H'|ε | 26|27 | {p,ε} | {j,g} |
| M | NM'' | 28 | {r,f,s,t} | {p,j,g} |
| M' | qNM'' | 53 | {q} | {p,j,g} |
| M'' | M'|ε | 29|30 | {q,ε} | {p,j,g} |
| N | rN|fHg|s|t | 31|32|33|34 | {r,f,s,t} | {q,p,j,g} |
| D | PA' | 35 | {z} | {c,l,g,m,o,d} |
| A' | PA'|QA'|ε | 36|37|38 | {z,u,x,y,ε} | {c,l,g,m,o,d} |
| L | B'RC' | 39 | {m,o,x,y} | {m,o,d,l,g} |
| B' | m|o|ε | 40|41|42 | {m,o,ε} | {x,y} |
| C' | QC'|ε | 43|44 | {u,x,y,ε} | {m,o,d,l,g} |
| Q | u|x|y | 45|46|47 | {u,x,y} | {u,x,y,m,o,d,l,g} |
| R | x|y | 48|49 | {x,y} | {u,x,y,m,o,d,l,g} |
| P | z | 50 | {z} | {z,u,x,y,l,c,g,m,o,d} |

## Tabuľka

Tabuľka 3 Prechodová tabuľka pre prvý prepis

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | e | f | g | h | i | j | k | l | m | o | p | r | s | t | q | u | x | y | z |
| S | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| B |  |  |  |  | 2 |  |  | 2 | 2 |  |  |  |  |  |  |  |  |  |  |  |  |  | 2 |
| X |  | 4 |  |  | 3 |  |  | 3 | 3 |  |  |  |  |  |  |  |  |  |  |  |  |  | 3 |
| C |  |  |  |  | 6 |  |  | 7 | 10 |  |  |  |  |  |  |  |  |  |  |  |  |  | 5 |
| Y |  |  |  | 9 |  |  |  |  |  |  | 8 |  |  |  |  |  |  |  |  |  |  |  |  |
| F |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 11 |
| Z |  |  |  |  |  |  | 13 |  |  |  |  | 12 |  |  |  |  |  |  |  |  |  |  |  |
| G |  |  |  |  |  | 14 |  |  |  |  |  |  | 14 | 14 |  |  |  |  |  |  | 14 | 14 | 14 |
| W |  |  |  |  |  |  | 16 |  |  |  |  | 15 |  |  |  |  |  |  |  |  |  |  |  |
| E |  |  |  |  |  | 17 |  |  |  |  |  |  | 17 | 17 |  |  |  |  |  |  | 17 | 17 | 17 |
| E' |  |  |  |  |  |  |  |  |  |  |  |  | 51 | 51 |  |  |  |  |  |  |  |  |  |
| E'' |  |  |  | 19 |  |  | 19 |  |  |  |  | 19 | 18 | 18 |  |  |  |  |  |  |  |  |  |
| K |  |  |  |  |  | 20 |  |  |  |  |  |  | 22 | 22 |  |  |  |  |  |  | 22 | 22 | 21 |
| J |  |  |  |  |  |  |  |  |  |  |  |  | 23 | 24 |  |  |  |  |  |  |  |  |  |
| H |  |  |  |  |  | 25 |  |  |  |  |  |  |  |  |  | 25 | 25 | 25 |  |  |  |  |  |
| H' |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 52 |  |  |  |  |  |  |  |  |
| H'' |  |  |  |  |  |  | 27 |  |  | 27 |  |  |  |  | 26 |  |  |  |  |  |  |  |  |
| M |  |  |  |  |  | 28 |  |  |  |  |  |  |  |  |  | 28 | 28 | 28 |  |  |  |  |  |
| M' |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 53 |  |  |  |  |
| M'' |  |  |  |  |  |  | 30 |  |  | 30 |  |  |  |  | 30 |  |  |  | 29 |  |  |  |  |
| N |  |  |  |  |  | 32 |  |  |  |  |  |  |  |  |  | 31 | 33 | 34 |  |  |  |  |  |
| D |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 35 |
| A' |  |  | 38 | 38 |  |  | 38 |  |  |  |  | 38 | 38 | 38 |  |  |  |  |  | 37 | 37 | 37 | 36 |
| L |  |  |  |  |  |  |  |  |  |  |  |  | 39 | 39 |  |  |  |  |  |  | 39 | 39 |  |
| B' |  |  |  |  |  |  |  |  |  |  |  |  | 40 | 41 |  |  |  |  |  |  | 42 | 42 |  |
| C' |  |  |  | 44 |  |  | 44 |  |  |  |  | 44 | 44 | 44 |  |  |  |  |  | 43 | 43 | 43 |  |
| Q |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 45 | 46 | 47 |  |
| R |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 48 | 49 |  |
| P |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 50 |

## Finálny prepis

Pred implementáciou sme sa rozhodli, že správnosť využitia pravidiel týkajúcich sa neterminálov "D" a "L" overíme už v lexikálnom analyzátore. Rozhodnutie vyplývalo hlavne z toho dôvodu, že z týchto neterminálov sa dá vygenerovať iba:

* slovo, ktoré môže obsahovať čísla (ale nesmie nimi začínať)
* číslo, ktoré nesmie začínať cifrou 0

Usúdili sme, že bude teda rozumnejšie a efektívnejšie overiť ich pri lexikálnom preklade a ďalej s výsledným slovom / číslom pracovať ako s jedným terminálom.

Tabuľka 4 Finálny prepis terminálov a neterminálov na nami zadefinované znaky

|  |  |  |  |
| --- | --- | --- | --- |
| Terminály | Znak | Neterminály | Znak |
| 'BEGIN' | a | program | S |
| 'END' | b | statement\_list | B |
| ':=' | c | statement | C |
| ';' | d | id\_list | F |
| 'READ' | e | expr\_list | G |
| '(' | f | expression | E |
| ')' | g | factor | K |
| 'WRITE' | h | op | J |
| 'IF' | i | bexpr | H |
| 'THEN' | j | bterm | M |
| 'ELSE' | k | bfactor | N |
| ',' | l | ident | D |
| '+' | m | number | L |
| '-' | o |  |  |
| 'OR' | p |  |  |
| 'NOT' | r |  |  |
| 'TRUE' | s |  |  |
| 'FALSE' | t |  |  |
| 'AND' | q |  |  |
| 'number\_term' | u |  |  |
|  |  |  |  |
|  |  |  |  |
| 'Ident\_term ' | z |  |  |

## Nedeterministickosť

Nedeterministickosť sa objavila pri niekoľkých pravidlách zobrazených v tabuľke nižšie.

Tabuľka 5 Zobrazenie neterminálov pri ktorých sme iedntifikovali nedeterministickosť

|  |  |  |  |
| --- | --- | --- | --- |
| Neterminály | Pravidlá | Spojené pravidlá + odtránenie ľavej rekurzie | Výsledky po faktorizácií |
| ~~E~~ | ~~TK~~ | ~~E -> EJK| K~~ | E-> KE’’ |
|  | ~~E -> KE’|K~~ | E‘‘->E‘|ε |
|  | ~~E‘->JKE‘|JK~~ | E‘->JKE‘‘ |
| ~~T~~ | ~~EJ| ε~~ |  |  |
|  | | | |
| ~~H~~ | ~~UM~~ | ~~H -> HpM|M~~ | H -> MH‘‘ |
|  | ~~H -> MH‘|M~~ | H‘‘-> H‘|ε |
|  | ~~H‘ -> pMH‘|pM~~ | H‘ -> pMH‘‘ |
| ~~U~~ | ~~Hp| ε~~ |  |  |
|  | | | |
| ~~M~~ | ~~AN~~ | ~~M -> MqN|N~~ | M -> NM‘‘ |
|  | ~~M -> NM‘|N~~ | M‘‘ -> M‘|ε |
|  | ~~M‘ -> qNM‘|qN~~ | M‘ -> qNM‘‘ |
| ~~A~~ | ~~Mq| ε~~ |  |  |

## First a Follow

Tabuľka 6 Finálne pravidlá, čísla paravidiel a vymenovanie first a follow

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Neterminály | Pravidlá | Čísla pravidiel | First | Follow |
| S | aBb | 1 | {a} | {$} |
| B | CX | 2 | {z,e,h,i} | {b} |
| X | CX|ε | 3|4 | {z,e,h,i,ε} | {b} |
| C | DcEd|efFgd|hfGgd|iHjCYd | 5|6|7|10 | {z,e,h,i} | {z,e,h,i,b,k,d} |
| Y | kC|ε | 8|9 | {k,ε} | {d} |
| F | DZ | 11 | {z} | {g} |
| Z | lDZ|ε | 12|13 | {l,ε} | {g} |
| G | EW | 14 | {f,z,m,o,u} | {g} |
| W | lEW|ε | 15|16 | {l,ε} | {g} |
| E | KE'' | 17 | {f,z,m,o,u} | {d,l,g} |
| E' | JKE'' | 51 | {m,o} | {d,l,g} |
| E'' | E'|ε | 18|19 | {m,o,ε} | {d,l,g} |
| K | fEg|D|L | 20|21|22 | {f,z,m,o,u} | {m,o,d,l,g} |
| J | m|o | 23|24 | {m,o} | {f,z,m,o,u} |
| H | MH'' | 25 | {r,f,s,t} | {j,g} |
| H' | pMH'' | 52 | {p} | {j,g} |
| H'' | H'|ε | 26|27 | {p,ε} | {j,g} |
| M | NM'' | 28 | {r,f,s,t} | {p,j,g} |
| M' | qNM'' | 53 | {q} | {p,j,g} |
| M'' | M'|ε | 29|30 | {q,ε} | {p,j,g} |
| N | rN|fHg|s|t | 31|32|33|34 | {r,f,s,t} | {q,p,j,g} |
| D | z | 35 | {z} | {c,l,g,m,o,d} |
| L | B'u | 36 | {m,o,u} | {m,o,d,l,g} |
| B' | m|o|ε | 37|38|39 | {m,o,ε} | {u} |

## Finálna prechodová tabuľka

Tabuľka 7 Prechodová tabuľka pre finálny prepis aj s defaultným terminálom x

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | e | f | g | h | i | j | k | l | m | o | p | r | s | t | q | u | z | x |
| S | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |
| B |  |  |  |  | 2 |  |  | 2 | 2 |  |  |  |  |  |  |  |  |  |  |  | 2 | 2 |
| X |  | 4 |  |  | 3 |  |  | 3 | 3 |  |  |  |  |  |  |  |  |  |  |  | 3 | ε |
| C |  |  |  |  | 6 |  |  | 7 | 10 |  |  |  |  |  |  |  |  |  |  |  | 5 | ε |
| Y |  |  |  | 9 |  |  |  |  |  |  | 8 |  |  |  |  |  |  |  |  |  |  | ε |
| F |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 11 | 11 |
| Z |  |  |  |  |  |  | 13 |  |  |  |  | 12 |  |  |  |  |  |  |  |  |  | ε |
| G |  |  |  |  |  | 14 |  |  |  |  |  |  | 14 | 14 |  |  |  |  |  | 14 | 14 | 14 |
| W |  |  |  |  |  |  | 16 |  |  |  |  | 15 |  |  |  |  |  |  |  |  |  | ε |
| E |  |  |  |  |  | 17 |  |  |  |  |  |  | 17 | 17 |  |  |  |  |  | 17 | 17 | 17 |
| E' |  |  |  |  |  |  |  |  |  |  |  |  | 51 | 51 |  |  |  |  |  |  |  | 51 |
| E'' |  |  |  | 19 |  |  | 19 |  |  |  |  | 19 | 18 | 18 |  |  |  |  |  |  |  | ε |
| K |  |  |  |  |  | 20 |  |  |  |  |  |  | 22 | 22 |  |  |  |  |  | 22 | 21 | ε |
| J |  |  |  |  |  |  |  |  |  |  |  |  | 23 | 24 |  |  |  |  |  |  |  | ε |
| H |  |  |  |  |  | 25 |  |  |  |  |  |  |  |  |  | 25 | 25 | 25 |  |  |  | 25 |
| H' |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 52 |  |  |  |  |  |  | 52 |
| H'' |  |  |  |  |  |  | 27 |  |  | 27 |  |  |  |  | 26 |  |  |  |  |  |  | ε |
| M |  |  |  |  |  | 28 |  |  |  |  |  |  |  |  |  | 28 | 28 | 28 |  |  |  | 28 |
| M' |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 53 |  |  | 53 |
| M'' |  |  |  |  |  |  | 30 |  |  | 30 |  |  |  |  | 30 |  |  |  | 29 |  |  | ε |
| N |  |  |  |  |  | 32 |  |  |  |  |  |  |  |  |  | 31 | 33 | 34 |  |  |  | ε |
| D |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 35 | 35 |
| L |  |  |  |  |  |  |  |  |  |  |  |  | 36 | 36 |  |  |  |  |  | 36 |  | 36 |
| B' |  |  |  |  |  |  |  |  |  |  |  |  | 37 | 38 |  |  |  |  |  | 39 |  | ε |

# Implementácia

V projekte sme vytvorili nasledujúce triedy:

* GrammarElement – abstraktná trieda obsahujúca dve hodnoty symbol a textovú reprezentáciu symbolu.
* Neterminal – Trieda reprezentujúca neterminál. Dedí od triedy GrammarElement. Obsahuje naviac metódu *replace*, ktorá na základe svojho symbolu vyvolá príslušné pravidlo z triedy *Pravidla*, ktoré nám vráti náhradu za tento neterminál podľa čakajúceho znaku(terminálu) na vstupe, privedeného argumentom. Každý neterminál obsahuje zoznam očakávaných terminálov na vstupe. To nám umožňuje poskytnúť presnejší výpis pri nájdení chyby.
* Terminal - Trieda reprezentujúca terminál. Dedí od triedy GrammarElement.
* Pravidla – Zoznam pravidiel. Pre každý neterminál máme vytvorenú jednu metódu, ktorá vracia pole terminálov a neterminálov získaných aplikovaním niektorého z pravidiel pre daný neterminál a terminál zo vstupu.
* Stack – Trieda reprezentujúca zásobník a vstup. Obsahuje potrebné metódy pre prácu s nimi akými sú napríklad *peek*, *pop*, *peekSecond –* používaná pri spamätaní z chýb*,* dve metódy pre jeho výpis(symboly alebo textovú reprezentáciu).
* Main – Obsahuje načítavanie zo vstupu, jeho prevedenie na terminály a uloženie do inštancie triedy *Stack,* reprezentujúcu vstup. Inicializovanie zásobníku. Zabezpečuje spracovanie vstupu spolu s spamätaním sa z chýb.

## Spamätanie sa z chýb

Identifikovali sme niekoľko prípadov, kedy sa môžeme zaseknúť. Vždy keď nastane jedna z nasledujúcich situácií tak používateľovi zobrazíme chybový výpis. Chybový výpis obsahuje informáciu o tom, ktorý z týchto scenárov nastal, očakávaný symbol na vstupe (získaný využitím prechodovej tabuľky) a symbol, ktorý sme dostali na vstupe. Ďalej ho informujeme o tom akým spôsobom sme túto situáciu vyriešili. V prílohe A sa nachádzajú ukážky všetkých chybových výpisov.

### Na vrchu zásobníka je terminál, ktorý nie je zhodný s terminálom na vstupe

V tomto prípade overíme, či sa nám terminál zo zásobníka nezhoduje s nasledujúcim terminálom na vstupe.

* V prípade, že áno, tak POPneme zo vstupu a pokračujeme ďalej.
* V prípade, že nie alebo, že na vstupe je iba jeden znak, tak POPneme zo zásobníku.

### Na vrchu zásobníka je neterminál bez aplikovateľného pravidla pre terminál na vstupe

V tomto prípade overíme, či v danom neterminály existuje pravidlo pre druhý znak na vstupe.

* V prípade, že áno, tak POPneme zo vstupu a pokračujeme ďalej.
* V prípade, že nie, tak aplikujeme defaultné pravidlo. Ak neterminál má iba jedno pravidlo, toto pravidlo je zvolené za defaultné. Ak neterminál má viac než jedno pravidlo, tak použijeme epsilon ako defaultné pravidlo.

# Príloha A Chybové výpisy

# Príloha B