# ProgettoLC parteA Alvise Bruniera Relazione

## Esercizio 1

## Esercizio 2

### Tabella di parsing

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| NT\la | ( | ) | + | , | = | > | and | id | not | num | or | pred | $ |
| A | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | A->  pred A' | 1 |
| A' | 3 | 10 | 4 | 5 | 12 | 12 | 10 | A'->EA'' | 12 | A'->EA'' | 10 | 3 | 1 |
| A'' | 3 | A''-> ε | 4 | A''->  ,EA'' | 6 | 6 | A''->  ε | 11 | 6 | 11 | A''->  ε | 6 | A''->  ε |
| B | 2 | 2 | 2 | 2 | 2 | 2 | 2 | B->EB' | 2 | B->EB' | 2 | B->A | 1 |
| B' | 3 | 3 | 13 | 3 | B'-> =E | B'-> >E | 8 | 14 | 3 | 14 | 8 | 3 | 1 |
| E | 3 | 9 | 4 | 3 | 3 | 3 | 16 | E-> id E' | 3 | E-> num E' | 16 | 3 | 1 |
| E' | 3 | E'->  ε | E'-> +E | E'->  ε | E'->  ε | E'->  ε | E'->  ε | 15 | 3 | 15 | E'->  ε | 3 | E'->  ε |
| P | P-> (PP' | 3 | 17 | 3 | 18 | 18 | 7 | P->  B | P-> not P | P->  B | 7 | P->  B | 1 |
| P' | 19 | 3 | 19 | 3 | 19 | 19 | P'-> and P) | 19 | 19 | 19 | P'-> or P) | 19 | 1 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | print "Incomplete program"; exit | | | | | | | |
| 2 | print "Internal error"; exit | | | | | | | |
| 3 | print "Unexpected {lookahead} "; skip lookahead | | | | | | | |
| 4 | print "Missing operand "; pop; push E'A'' | | | | | | | |
| 5 | print "Missing expression "; pop; push A'' | | | | | | | |
| 6 | print "Cannot use {lookahead} in predicate"; remove until ", ) and or $" excluded | | | | | | | |
| 7 | print "Missing operand "; pop; push P'; | | | | | | | |
| 8 | print "Incomplete compare "; pop | | | | | | | |
| 9 | print "Incomplete expression "; pop | | | | | | | |
| 10 | print "Empty predicate "; pop | | | | | | | |
| 11 | print "Missing comma "; push E | | | | | | | |
| 12 | pop; push A'' | | | | | | | |
| 13 | print "Missing second operand "; skip lookahead | | | | | | | |
| 14 | print "Missing operator"; pop; push E | | | | | | | |
| 15 | print "Missing operand"; pop; push E | | | | | | | |
| 16 | print "Missing expression"; pop | | | | | | | |
| 17 | print "Missing pred"; pop; push E'A'' | | | | | | | |
| 18 | print "Missing operand"; pop; push B' | | | | | | | |
| 19 | print "Missing operator"; pop; push P | | | | | | | |

### Tabella dei mismatch sui non terminali

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| st\la | ( | ) | + | , | = | > | and | id | not | num | or | pred | $ |
| ( | acc | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| ) | 4 | acc | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 1 |
| + | 2 | 2 | acc | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| , | 2 | 2 |  | acc | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| = | 2 | 2 | 2 | 2 | acc | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| > | 2 | 2 | 2 | 2 | 2 | acc | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| and | 2 | 2 | 2 | 2 | 2 | 2 | acc | 2 | 2 | 2 | 2 | 2 | 2 |
| id | 2 | 2 | 2 | 2 | 2 | 2 | 2 | acc | 2 | 2 | 2 | 2 | 2 |
| not | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | acc | 2 | 2 | 2 | 2 |
| num | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | acc | 2 | 2 | 2 |
| or | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | acc | 2 | 2 |
| pred | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | acc | 2 |
| $ | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | halt |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | print "Incomplete program"; exit | | | | | | | | |
| 2 | print "Internal error"; exit | | | | | | | | |
| 3 | print "Unexpected {lookahead} "; skip lookahead | | | | | | | | |
| 4 | print "Missing )"; pop | | | | | | | | |

### Esecuzione su una stringa d’esempio

|  |  |  |
| --- | --- | --- |
| $P | ( ( pred and id + num > num ) ) or ( id not = id and pred num , ) )$ | P->(PP' |
| $P'P( | ( ( pred and id + num > num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P'P | ( pred and id + num > num ) ) or ( id not = id and pred num , ) )$ | P->(PP' |
| $P'P'P( | ( pred and id + num > num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P'P'P | pred and id + num > num ) ) or ( id not = id and pred num , ) )$ | P->B |
| $P'P'B | pred and id + num > num ) ) or ( id not = id and pred num , ) )$ | B->A |
| $P'P'A | pred and id + num > num ) ) or ( id not = id and pred num , ) )$ | A->pred A' |
| $P'P'A' pred | pred and id + num > num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P'P'A' | and id + num > num ) ) or ( id not = id and pred num , ) )$ | Empty predicate; pop |
| $P'P' | and id + num > num ) ) or ( id not = id and pred num , ) )$ | P'-> and P) |
| $P')P and | and id + num > num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P')P | id + num > num ) ) or ( id not = id and pred num , ) )$ | P->B |
| $P')B | id + num > num ) ) or ( id not = id and pred num , ) )$ | B->EB' |
| $P')B'E | id + num > num ) ) or ( id not = id and pred num , ) )$ | E->id E' |
| $P')B'E' id | id + num > num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P')B'E' | + num > num ) ) or ( id not = id and pred num , ) )$ | E'->+E |
| $P')B'E+ | + num > num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P')B'E | num > num ) ) or ( id not = id and pred num , ) )$ | E->num E' |
| $P')B'E' num | num > num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P')B'E' | > num ) ) or ( id not = id and pred num , ) )$ | E->ε |
| $P')B' | > num ) ) or ( id not = id and pred num , ) )$ | B'-> >E |
| $P')E> | > num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P')E | num ) ) or ( id not = id and pred num , ) )$ | E-> num E' |
| $P')E' num | num ) ) or ( id not = id and pred num , ) )$ | acc |
| $P')E' | ) ) or ( id not = id and pred num , ) )$ | E->ε |
| $P') | ) ) or ( id not = id and pred num , ) )$ | acc |
| $P' | ) or ( id not = id and pred num , ) )$ | Unexpected ); skip ) |
| $P' | or ( id not = id and pred num , ) )$ | P'-> or P) |
| $)P or | or ( id not = id and pred num , ) )$ | acc |
| $)P | ( id not = id and pred num , ) )$ | P->(PP' |
| $)P'P( | ( id not = id and pred num , ) )$ | acc |
| $)P'P | id not = id and pred num , ) )$ | P->B |
| $)P'P | id not = id and pred num , ) )$ | B->EB' |
| $)P'B'E | id not = id and pred num , ) )$ | E->id E' |
| $)P'B'E' id | id not = id and pred num , ) )$ | acc |
| $)P'B'E' | not = id and pred num , ) )$ | Unexpected not; skip not |
| $)P'B'E' | = id and pred num , ) )$ | E'->ε |
| $)P'B' | = id and pred num , ) )$ | B'-> =E |
| $)P'E= | = id and pred num , ) )$ | acc |
| $)P'E | id and pred num , ) )$ | E-> id E' |
| $)P'E' id | id and pred num , ) )$ | acc |
| $)P'E' | and pred num , ) )$ | E'->ε |
| $)P' | and pred num , ) )$ | P'-> and P) |
| $))P and | and pred num , ) )$ | acc |
| $))P | pred num , ) )$ | P->B |
| $))B | pred num , ) )$ | B->A |
| $))A | pred num , ) )$ | A-> pred A' |
| $))A' pred | pred num , ) )$ | acc |
| $))A' | num , ) )$ | A'->EA'' |
| $))A''E | num , ) )$ | E-> num E' |
| $))A''E' num | num , ) )$ | acc |
| $))A''E' | , ) )$ | E'->ε |
| $))A'' | , ) )$ | A''->,EA'' |
| $))A''E, | , ) )$ | acc |
| $))A''E | ) )$ | Incomplete expression; pop |
| $))A'' | ) )$ | A''->ε |
| $)) | ) )$ | acc |
| $) | )$ | acc |
| $ | $ | halt |