**REGULAR DEFINITION**

|  |  |
| --- | --- |
| Name | Definition |
| upperChar | { A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z } |
| lowerChar | { a, b, c, d, e, f, g, h, I, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z } |
| zero | { 0 } |
| num | {1, 2, 3, 4, 5, 6, 7, 8, 9} |
| negaNum | {-1, -2, -3, -4, -5, -6, -7, -8, -9} |
| arithOp | { +, -, \*, / } |
| logicOp | { && , || } |
| RelOp | { <, >, <=, =>, ==, != } |
| EqualOp | { = } |
| ASCII | Any printable character. |
| Bool | { AFFIRMATIVE, NEGATIVE } |
| delim1 | { =, ; , space } |
| delim2 | { (, space } |
| delim3 | { ; , space } |
| delim4 | { : , space } |
| delim5 | { space } |
| delim6 | { =, , , [, space } |
| delim7 | { {, newline, space } |
| delim8 | {=, , , }, space } |
| delim9 | { <, space, ;,” } |
| delim10 | { =,. , ; , space } |
| del1 | { lowerChar, zero, num, (, space } |
| del2 | { lowerChar, newline, upperChar, space } |
| del3 | { lowerChar, xnewline, space } |
| del4 | { lowerChar, zero, num, (, negaNum, space } |
| del5 | { lowerChar, zero, num, (, space } |
| del6 | { lowerChar, upperChar, }, newline, num, zero, space, ‘, “ } |
| del7 | { newline, ;, }, comma, upperChar, lowerChar, space } |
| del8 | { lowerChar, upperChar, num, zero, }, negaNum, “, (, space } |
| del9 | { ), {, ;, arithOp, logicOp, RelOp, space } |
| del10 | { lowerChar, num, zero, space } |
| del11 | { [, ;, ), =, space } |
| del12 | { lowerChar, upperChar, zero, num, {, ‘, “, space } |
| del13 | { lowerChar, num, zero, space, ‘,” } |
| del14 | { “, lowerChar, space } |
| del15 | { lowerChar, ), ;, num, space } |
| del16 | { num, zero, “, ‘, lowerChar, space } |
| term1 | { armop, logop, relop, ;, ], }, comma, :, space } |
| term2 | { armop, logop, relop, ;, ), }, comma, :, ., space } |
| term3 | { ;, }, comma, space } |
| term4 | { ;, }, comma, ), space } |
| term5 | { =, comma, armop, logop, relop, (, ), [, ], }, ;, space } |
| term6 | { space, newline } |

**RESERVED WORDS**

|  |  |  |  |
| --- | --- | --- | --- |
| Words | C | Regular Expression | Token |
| PrimaryMission | Main | (P) (r) (i) (m) (a) (r) (y) (M) (i) (s) (s) (i) (o) (n) | PrimaryMission |
| commence | clrscr | (c) (o) (m) (m) (e) (n) (c) (e) | commence |
| post | printf | (p) (o) (s) (t) | post |
| captured | scanf | (c) (a) (p) (t) (u) (r) (e) (d) | captured |
| hold | const | (h) (o) (l) (d) | hold |
| backup | return | 1. (a) (c) (k) (u) (p) | backup |
| campaign | switch | (c) (a) (m) (p) (a) (i) (g) (n) | campaign |
| operation | case | (o) (p) (e) (r) (a) (t) (i) (o) (n) | operation |
| abort | break | (a) (b) (o) (r) (t) | abort |
| unit | int | (u) (n) (i) (t) | unit |
| company | char | (c) (o) (m) (p) (a) (n) (y) | company |
| digit | float | (d) (i) (g) (i) (t) | digit |
| inquire | for | (i) (n) (q) (u) (i) (r) (e) | inquire |
| inorder | if | (i) (n) (o) (r) (d) (e) (r) | Inorder |
| otherorder | else if | (o) (t) (h) (e) (r) (o) (r) (d) (e) (r) | otherorder |
| order | else | (o) (r) (d) (e) (r) | order |
| go | do | (g) (o) | go |
| phase | while | (p) (h) (a) (s) (e) | phase |
| miss | void | (m) (i) (s) (s) | miss |
| orbat | goto | (o) (r) (b) (a) (t) | orbat |
| auto | default | (a) (u) (t) (o) | auto |
| deploy | getch | (d) (e) (p) (l) (o) (y) | deploy |

**RESERVED SYMBOLS**

|  |  |  |
| --- | --- | --- |
| Reserved Symbols | Regular Expression | Token |
| + | (+) | + |
| - | (-) | - |
| \* | (\*) | \* |
| / | (/) | / |
| % | (%) | % |
| = | (=) | = |
| ; | (;) | ; |
| . | (.) | . |
| && | (&) (&) | && |
| || | (|) (|) | || |
| ! | (!) | ! |
| @ | (@) | @ |
| // | (/) (/) | // |
| ++ | (+) (+) | ++ |
| -- | (-) (-) | -- |
| += | (+) (=) | += |
| -= | (-) (=) | -= |
| \n | (\) (n) | \n |
| != | (!) (=) | != |