

DIGIT [0-9]

ID [a-z][a-z0-9]*

%%

{DIGIT}+ printf("An integer: %s (%d)\n", yytext,atoi(yytext));

"+" printf("Operator : %s \n", yytext);

"-" printf("Operator : %s \n", yytext);

"/" printf("Operator : %s \n", yytext);

"=" printf("Operator : %s \n", yytext);

"*" printf("Operator : %s \n", yytext);

"(" printf("Separator : %s \n", yytext);

")" printf("Separator : %s \n", yytext);

"{" printf("Separator : %s \n", yytext);

"}" printf("Separator : %s \n", yytext);

"[" printf("Separator : %s \n", yytext);

"]" printf("Separator : %s \n", yytext);

":" printf("Separator : %s \n", yytext);

"," printf("Separator : %s \n", yytext);

"," printf("Separator : %s \n", yytext);

"number" printf("Reserved word : %s \n", yytext);

"str" printf("Reserved word : %s \n", yytext);

"char" printf("Reserved word : %s \n", yytext);

"bool" printf("Reserved word : %s \n", yytext);

"list" printf("Reserved word : %s \n", yytext);

"multiple" printf("Reserved word : %s \n", yytext);

"sum" printf("Reserved word : %s \n", yytext);

```

"of"          printf("Reserved word : %s \n", yytext);
"diff"        printf("Reserved word : %s \n", yytext);
"div"         printf("Reserved word : %s \n", yytext);
"greater"     printf("Reserved word : %s \n", yytext);
"lower"       printf("Reserved word : %s \n", yytext);
"than"        printf("Reserved word : %s \n", yytext);
"let"         printf("Reserved word : %s \n", yytext);
"be"          printf("Reserved word : %s \n", yytext);
"show"        printf("Reserved word : %s \n", yytext);
"read"        printf("Reserved word : %s \n", yytext);
"is"          printf("Reserved word : %s \n", yytext);
"START"       printf("Reserved word : %s \n", yytext);
"END"         printf("Reserved word : %s \n", yytext);
"if"          printf("Reserved word : %s \n", yytext);
"else"        printf("Reserved word : %s \n", yytext);
"then"        printf("Reserved word : %s \n", yytext);
"for"         printf("Reserved word : %s \n", yytext);
"from"        printf("Reserved word : %s \n", yytext);
"do"          printf("Reserved word : %s \n", yytext);
"until"       printf("Reserved word : %s \n", yytext);
"done"        printf("Reserved word : %s \n", yytext);

```

```

{ID}          printf( "An identifier: %s\n", yytext );

```

```

%%

```

```

main( argc, argv )

```

```

int argc;

```

```

char **argv;

```

```
{  
    ++argv, --argc;    /* skip over program name */  
    if ( argc > 0 )  
        yyin = fopen( argv[0], "r" );  
    else  
        yyin = stdin;  
    yylex();  
}
```

```
#ifndef yywrap  
    yywrap() { return 1; }  
#endif
```