

# Compiladores

Prof. Ricardo

[ricardo.martins@udesc.br](mailto:ricardo.martins@udesc.br)

---

```
delim    [ \t]
ws        {delim}+
digito    [0-9]
num       {digito}+(\.{digito}*(E[+-]?{digito}+)?)?
```

```
%%
```

```
{ws}      {}
```

```
"+" {return TADD;}
```

```
"-" {return TSUB;}
```

```
"*" {return TMUL;}
```

```
"/" {return TDIV;}
```

```
"(" {return TAPAR;}
```

```
")" {return TFPAR;}
```

```
\n {return TFIM;}
```

```
{num}     {yyval=atof(yytext); return TNUM;}
```

# yacc

---

```
%{  
#include <stdio.h>  
#include <stdlib.h>  
#define YYSTYPE double  
%}  
  
%token TADD TMUL TSUB TDIV TAPAR TFPAR TNUM TFIM  
  
%%
```

# Yacc/Bison

---

```
Linha :Expr TFIM {printf("Resultado:%lf\n", $1);exit(0);}
;
Expr: Expr TADD Termo {$$ = $1 + $3;}
    | Expr TSUB Termo {$$ = $1 - $3;}
    | Termo
;
Termo: Termo TMUL Fator {$$ = $1 * $3;}
    | Termo TDIV Fator {$$ = $1 / $3;}
    | Fator
;
Fator: TNUM
    | TAPAR Expr TFPAR {$$ = $2;}
;
%%
```

# Yacc/Bison

---

```
int yyerror (char *str)
{
    printf("%s - antes %s\n", str, yytext);
}
```

```
int yywrap()
{
    return 1;
}
```

# Programa

---

```
#include <stdio.h>
```

```
extern FILE *yyin;
```

```
int main()  
{  
    yyin = stdin;  
    printf("Digite uma expressão:");  
    yyparse();  
    return 0;  
}
```