

LEX FILE

```
%%  
[0-9]+ {return NUM;}  
\+ return PLUS;  
\- return MINUS;  
%%
```

YACC FILE

```
%{  
#include<stdio.h>  
#include<string.h>  
#include<stdlib.h>  
typedef struct node  
{  
    char *token;  
    struct node *left;  
    struct node *right;  
} node;  
node *mknode(char *token, node *left, node *right);  
void printtree(node *tree);  
#define YYSTYPE struct node*  
%}  
%token NUM,PLUS,MINUS  
%left PLUS MINUS  
%%  
s: exp { printf("OK\n"); printtree($1); }  
exp: exp PLUS exp {$$=mknode("+",$1,$3);}  
    | exp MINUS exp {$$=mknode("-",$1,$3);}  
    | NUM {$$=mknode(yytext,NULL,NULL)};  
%%  
#include "lex.yy.c"  
main()  
{  
    return yyparse();  
}
```

```

node *mknode(char *token,node *left,node *right)
{
    node *newnode=(node*)malloc(sizeof(node));
    char *newstr=(char*)malloc(sizeof(token) + 1);
    strcpy(newstr,token);
    newnode->left=left;
    newnode->right=right;
    newnode->token=newstr;
    return newnode;
}
void printtree(node *tree)
{
    printf("%s\n",tree->token);
    if(tree->left)
        printtree(tree->left);
    if(tree->right)
        printtree(tree->right);
}
int yyerror()
{
    printf("MY ERROR\n");
    return 0;
}

```

TEXT FILE 1

2 + 3 – 4

TEXT FILE 2

1 + 2 + 3 + 4 + 5