

Parse table:

	eof	+	-	*	/	()	id	num
Expr						$E \rightarrow TE'$		$E \rightarrow TE'$	$E \rightarrow TE'$
Expr'	$E' \rightarrow \epsilon$	$E' \rightarrow +TE'$	$E' \rightarrow -TE'$				$E' \rightarrow \epsilon$		
Term						$T \rightarrow FT'$		$T \rightarrow FT'$	$T \rightarrow FT'$
Term'	$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$	$T' \rightarrow /FT'$		$T' \rightarrow \epsilon$		
Factor						$F \rightarrow (E)$		$F \rightarrow id$	$F \rightarrow num$

Table-driven parser:

```

push the start symbol, S, onto stack;
focus ← top of Stack;
loop forever;
  if (focus = eof and word = eof)
    then report success and exit the loop;
  else if (focus ∈ T or focus = eof) then begin;
    if focus matches word then begin;
      pop Stack;
      word ← NextWord();
    end;
    else report an error looking for symbol at top of stack;
  end;
  else begin; /* focus is a nonterminal */
    if Table[focus,word] is  $A \rightarrow B_1B_2 \dots B_k$  then begin;
      pop Stack;
      for i ← k to 1 by -1 do;
        if ( $B_i \neq \epsilon$ )
          then push  $B_i$  onto Stack;
        end;
      end;
    else report an error expanding focus;
  end;
  focus ← top of Stack;
end;

```