VBC → VALOR BINAIRE CALCULÉ → CALCOLAZIONE DI ESPRESSIONI MATEMATICHE. DEVE RISPETTARE LA PRECEDENZA DELLE OPERAZIONI

USIAMO IL **PARSING RICORSIVO DISCENDENTE** che implementa naturalmente la precedenza.

EXPR() ← ADDIZIONI (precedenza bassa)
↑
TERM() ← MOLTIPLICAZIONI (precedenza media)
↑
FACTOR() ← numeri e parentesi (precedenza alta)

FACTOR() → LEGGE L'ELEMENTO più piccolo (NUM SINGOLO o INTERA ESPRESSIONE TRA parentesi)

- Se TROVI '(' ENTRI RICORSIVAMENTE in EXPR() PER VALUTARE TUTTO QUELLO CHE C'È DENTRO.

ex: (3+4) → '(' → EXPR() → SVOLGE 3+4=7 → CONTROLLA ')' E POI ESCE

TERM() → GESTISCE LE MOLTIPLICAZIONI CONSECUTIVE
→ LEGGE UN FACTOR(), POI SE TROVA '*' LEGGE IL PROSSIMO FACTOR() E MOLTIPLICA

EXPR() → È IL LIVELLO più ALTO. LEGGE UN term(). SE TROVA '+' LEGGE IL PROSSIMO term() E ADDIZIONA

Ex → 3+4*5 → EXPR() chiama term(). TERM() legge 3 (factor) non trova '*' e RITORNA 3. EXPR() vede '+' CHIAMA DI NUOVO term(). TERM() legge 4, TROVA '*' E MOLTIPLICA PER 5. EXPR() ADDIZIONA = 23

LA CHIAVE
LA RICORSIONE
expr() chiama term()
term() chiama factor()
QUANDO expr() INCONTRA (+)
ASPETTA che term() processi tutte le moltiplicazioni prima di proseguire.

```c
#include <stdlib.h>  #include <ctype.h>  #include <stdio.h>

char *s;   // puntatore globale che scorre input

void error(char c){
  if (c) printf("Unexpected token '%c'\n");   // c è il carattere errato
  else   printf("Unexpected end of input\n");  // c == 0 (fine stringa)
  exit(1); // termina come errore
}

int expr();
int term();
int factor();

int factor(){
  if(isdigit(*s)) return(*s++ - '0');   // se il carattere corrente è una cifra convertilo in numero e sposta il puntatore next
  if(*s == '('){                        // se il carattere corrente è '('
    s++;                                // salta '('
    int value = expr();                 // chiama expr() x valutare
    if(*s != ')') error(*s);            // se il carattere corrente non è ')' errore → parent. non chiusa
    s++;                                // sposta puntatore oltre ')'
    return(value);                      // ritorna val calcolato
  }
  error(*s); // altrimenti errore - carattere inaspettato
}
return(0);

int term(){
  int value = factor();                 // leggi il primo numero tramite factor() e conserva il risultato in value
  while(*s == '*'){                     // mentre
    s++;                                // sposta puntatore oltre '*'
    value = value * factor();           // leggi prox numero con factor e moltiplica value con il nuovo numero conservando il risultato in value
  }
  return(value);                        // ritorna il valore finito
}

int expr(){
  int value = term();                   // leggi il primo termine e conserva il valore in value
  while(*s == '+'){                     // mentre c'è +
    s++;                                // salta +
    value = value + term();             // aggiorna il valore leggendo il prox termine e sommandolo al valore precedente
  }
  return(value);
}

int main(int ac, char **av){
  if(ac != 2) return(1);
  s = av[1];                            // valorizzo la variabile globale con il primo argomento (la stringa da valut.)
  int result = expr();                  // valuta tutto chiamando expr()
  if(*s) error(*s);
  printf("%d\n", result);
  return(0);
}
```

"2+3*4"

MAIN: S punta a "2+3*4"

↳ chiama expr() → S punta a "2+3*4"

chiama term() → S punta a "2+3*4"

① → chiama factor()

LEGGE '2', RITORNA 2 → S ora punta a "+3*4"

② → TERM() torna con VALUE = 2

③ → vede + (non *) ed esce dal while

RITORNA ②

S punta a "+3*4"

expr() DUNQUE TORNA con VALUE 2

vede + → entra nel while

sposta S oltre il +

S ora punta a "3*4"

chiama di nuovo term()

S punta sempre a "3*4" e

Quindi chiama factor() che legge ③,

RITORNA 3 ora S punta a "*4"

vede '*' ed entra nel while.

sposta S oltre "*" ora S punta a 4

Valore = 3 * 4 = 12

vede fine stringa (non *) esce dal while

RITORNA ⑫

expr()

continua con valore ② + ⑫ = 14 vede fine stringa (non '+') ed esce dal while RITORNA ⑭

MAIN() → RICEVE 14 Se è a fine stringa e STAMPA 14