

Evaluador de expresiones

- Características de las expresiones:
 - Operandos: números enteros (1 dígito)
 - Operadores: +, -, *, /
 - Paréntesis para agrupar
 - La expresión se va a almacenar en una cadena de caracteres
- Subproblemas:
 1. Validar los paréntesis
 2. Convertir la expresión a notación postfija
 3. Evaluar la expresión postfija

Convertir la expresión a notación postfija

- ❖ Inicializar una pila de caracteres
- ❖ Mientras el siguiente carácter de la expresión no sea fin de cadena
 - Si c es '(' entonces meter c en la pila
 - Si c es ')' entonces sacar elementos de la pila y agregarlos a la cadena postfija, hasta que se haya sacado un paréntesis izquierdo (éste último no se guarda)
 - Si c es un operando (número $isdigit(c)$), agregar el operando a la cadena postfija
 - Si c es un operador ($c == '+' ||$), entonces repetir lo siguiente hasta que el operador c se meta en la pila
 - Si la pila está vacía o el operador c tiene una prioridad mayor que la prioridad del elemento tope de la pila, entonces
 - ◆ Mete c a la pila (salir del ciclo)
 - En otro caso sacar un operador de la pila y agregarlo a la cadena postfija
- ❖ Sacar todos los elementos restantes de la pila y agregarlos a la cadena postfija

Prioridad:

(tiene prioridad	0
+, -	tienen prioridad	1
*, /	tienen prioridad	2

Evaluar la expresión postfija

- ❖ Iniciar una pila de enteros
- ❖ Recorrer la cadena *exp* de izquierda a derecha, agregando el siguiente carácter a *car* y mientras no es fin de cadena:
 - Si *car* es un operando (número) entonces mete el valor *entero* del operando a la pila: (guardarlo en una cadena de caracteres (*num*) y convertir la cadena de caracteres *num* a su valor numérico)
 - Si *car* es un operador, entonces sacar dos operandos de la pila (*op2*, *op1*) aplicar el operador a los operandos ($r = op1 \text{ car } op2$). Meter el resultado r a la pila
 - Si *car* es un espacio, seguir
- ❖ Regresar como resultado el *tope*