

E. T. S. DE INGENIERÍA INFORMÁTICA
Departamento de Estadística, I.O. y Computación
Teoría de Autómatas y Lenguajes Formales

Considérese la siguiente declaración para la clase pila:

```

template <class Tipo> class Stack {
    int maxStack;
    int sp;
    Tipo *stk;
public:
    Stack(int size);           // Constructor
    ~Stack();                 // Destructor
    int push(Tipo& a);        // Inserta un elemento en la pila
    Tipo pop();               // Devuelve el elemento que está en
                             // la cima de la pila
    int empty();              // Comprueba si la pila está vacía
    int full();               // Comprueba si la pila está llena
};

```

Se pide:

1. Implementar los métodos de la clase.
2. Utilizar la clase anterior para resolver los siguientes problemas:
 - Comprobar si una expresión de corchetes y paréntesis que se introduce desde archivo está correctamente parentizada.

```

[ ( ) ] // es correcto
[ ( ] ) // no es correcto

```

- Evaluar una expresión en notación postfija que se introduce desde archivo. La expresión $12 + 15$ quedaría en el archivo en notación postfijo del siguiente modo:

```

OPERANDO 12
OPERANDO 15
OPERADOR +
FIN

```

- La entrada de datos se hará desde la línea de comandos según el siguiente formato, `programa -op <codigo_operacion> <fich_in>`, mostrando un mensaje de modo de uso en caso de uso incorrecto.

Recuerda que las siguientes indicaciones son de carácter obligatorio en el desarrollo de tus programas.

- Un identificador SÓLO puede escribirse todo con mayúsculas si se trata de una constante. Esto es válido incluso si el identificador sólo consta de un único carácter (por otra parte, eso no debería suceder, porque no deberíamos utilizar identificadores de un único carácter salvo en situaciones muy concretas).
- Comentarios en la declaración de variables y en las definiciones de clases.
- Identificadores significativos para las variables.