

Tarea #3

Normalmente, las personas se reúnen en sitios comunes a comer, discutir, estudiar, entre otros. Eloy, por otro lado, va a los sitios comunes a pensar como sería si el mundo tuviera solo 2 notaciones para las expresiones, dejando a un lado la notación infija, que todo el mundo conoce.

A Eloy le gusta mucho la notación postfija de las expresiones, tanto así que comienza a transformar todas las expresiones de su cuaderno de matemáticas para que todas aparezcan escritas en notación postfija. Por desgracia, Eloy se equivocó de cuaderno, y transformó las expresiones del cuaderno de Samuel, a quien no le gusta para nada la notación postfija de las expresiones.

A Samuel le encanta la notación prefija de las expresiones, y es capaz de transformar expresiones de notación infija a prefija, pero no de notación postfija a prefija. Por esto él pide tu ayuda, ya que es una tarea que sólo un excelente programador podría hacer, te dirá todas las expresiones del cuaderno de Eloy y tu tarea es hallar el árbol de expresión, con el fin de que Samuel pueda aprender la transformación que no sabe hacer.

Datos de entrada:

La primera línea contendrá un entero T , que representa el número de casos de prueba a evaluar. Cada caso de prueba comenzará con un entero M , que significará el grado del árbol a construir (ie., este árbol será M -ario). Seguido a esto, una línea con la expresión en notación postfija, leída del cuaderno de Samuel, dicha expresión no contendrá más de 10000 caracteres. Debe asumir que todos los operandos serán de un solo dígito, y los operadores a utilizar sólo serán '+' y '*'. Mas detalles de la entrada en el ejemplo. *La entrada será dada por standard input.*

Datos de salida:

Por cada caso de prueba, usted deberá imprimir en preorden el árbol M -ario generado a partir de la expresión postfija leída del cuaderno de Samuel, indicando, por cada vértice, el operando u operador, la altura y el id de su conjunto de hermanos; la salida debe llevar el siguiente formato: "O/A/Id", donde O es el operando u operador, A es la altura del vértice (comenzando en 0) y Id es el id del vértice dentro de su conjunto de hermanos (Id deberá ser "R" cuando se trate de la raíz del árbol, más detalles sobre esto en el ejemplo). *La salida debe ser dada por standard output.*

Ejemplo:

Entrada	Salida
3	+ / 0 / R 4 / 1 / 0 9 / 1 / 1
2	* / 0 / R 4 / 1 / 0 + / 1 / 1 3 / 2 / 0 * / 2 / 1 5 / 3 / 0 2 / 3 / 1
49+	+ / 0 / R 2 / 1 / 0 + / 1 / 1 5 / 2 / 0 4 / 2 / 1 5 / 2 / 2 * / 1 / 2 1 / 2 / 0 2 / 2 / 1 3 / 2 / 2
2	
4352*+*	
3	
2545+123*+	

Límites:

$$1 \leq T \leq 10$$

$$2 \leq M \leq 50$$

Recomendaciones, condiciones de entrega y otros aspectos:

- El tiempo límite de ejecución es de 2 segundos, cualquier programa que no dé respuesta en ese periodo de tiempo será detenido y calificado con la nota mínima.
- La tarea es de carácter individual.
- Los lenguajes permitidos son: C, C++, Java6.
- Los programas entregados serán corregidos con otro programa corrector, por lo que deben seguir al pie de la letra el formato de entrada y el de salida (incluyendo espacios), indicado en el enunciado. Cualquier programa que no lea correctamente la entrada o de cómo resultado una salida en un formato incorrecto automáticamente serán calificados con la nota mínima.
- La fecha límite de entrega es el miércoles 21/11/2012. A más tardar ese día deberá enviar un correo con el código fuente a su respectivo preparador. El asunto del correo deberá ser "Tarea #3, MD3", y en el correo deberá indicar su nombre, su cédula y su sección.
- Las copias serán severamente penalizadas según lo establecido en la Ley de Universidades. Se anima a la discusión pero se prohíbe la copia de proyectos. Cualquier tarea entregada debe ser fruto de su propio trabajo.
- Luego de la corrección, si no está de acuerdo con la nota obtenida, se podrá fijar una fecha para la revisión de su programa.

Luis Arguello, Noviembre 2012.