

Tarea 1:

Listas, filas y colas

Dr. Miguel Romero,
Dra.(c) Martita Muñoz

plazo de entrega: 22/04/2022

Instrucciones generales. La tarea puede ser desarrollada por un máximo de 3 personas. Debe proporcionar un informe (20 pts.), de no más de dos hojas por problema, que describa la solución algorítmica que ha implementado e indique el orden de magnitud de los algoritmos.

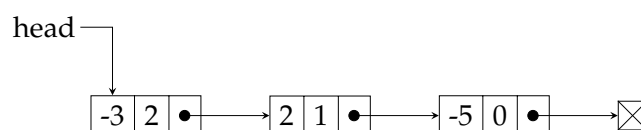
Problema 1 Clase Polinomio

(40 pts.)

Se requiere implementar una clase en java para representar polinomios de modo que permita realizar las siguientes operaciones (métodos):

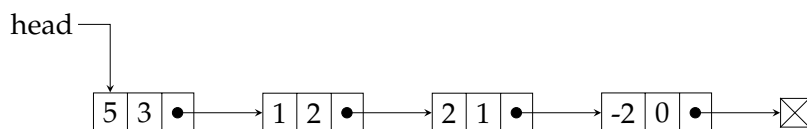
- Crear un polinomio vacío.
- Agregar un término al polinomio.
- Convertir un objeto de tipo polinomio a string.
- Sumar dos polinomios.
- Multiplicar dos polinomios.
- Dado un valor, evaluar el polinomio.
- Indicar el grado del polinomio.

Para la implementación se debe utilizar una lista enlazada donde cada nodo almacene el coeficiente, el exponente del término y un puntero al siguiente nodo. Por ejemplo: $g(x) = -3x^2 + 2x - 5$, se representaría en una lista enlazada como:



La lista enlazada debe estar ordenada por coeficiente de mayor a menor en todo momento. Además, no pueden haber dos nodos con igual exponente. Si

en algún paso intermedio ocurre, se deben reducir en un solo término. por ejemplo: si tenemos dos polinomios $g(x) = 2x^3 + x$; $p(x) = 3x^3 + x^2 + x - 2$ y sumamos $g(x) + p(x)$, un resultado parcial como: $2x^3 + 3x^3 + x^2 + x + x - 2$ debería reducirse a $5x^3 + x^2 + 2x - 2$. Lo que se representaría en la lista enlazada como:



Problema 2 Transbordador

(40 ptos.)

En aquellos lugares donde no existen puentes, es común que existan pequeños transbordadores o "balsas" para transportar automóviles a través de los ríos. Los transbordadores fluviales, a diferencia de sus primos más grandes, funcionan con una línea guía y funcionan con la corriente del río. Los automóviles ingresan al transbordador desde un extremo, el transbordador cruza el río y los automóviles salen por el otro extremo del transbordador (ver figura 1, o visite: <https://twitter.com/toprosur/status/1039834117221826561?lang=mr>).



Figure 1: Balsa Copihue, río Tolten región de la Araucanía, Chile

Hay un transbordador que cruza el río y puede llevar n automóviles a través del río en t minutos y regresar en t minutos. Un automóvil puede llegar a cualquiera de las orillas del río para ser transportado por el transbordador a la orilla opuesta.

El transbordador viaja continuamente de un lado a otro entre las orillas siempre que lleve un automóvil o haya al menos un automóvil esperando en cualquiera de las orillas. Cada vez que el transbordador llega a uno de los extremos, bajan los vehículos que lleva y carga hasta n vehículos que están esperando para cruzar. Si hay más de n , se cargan los que llevan más tiempo esperando. Si no hay coches esperando en ninguna de las orillas, el transbordador espera a que llegue uno, lo carga (si llega a la misma orilla del transbordador) y cruza el río. ¿A qué hora llega cada automóvil al otro lado del río?

Formato de entrada

Los datos se deben leer desde la entrada estandar (uso de Scanner en java).

La primera línea de entrada contiene un entero c , que indica el número de casos de prueba. Cada caso de prueba comienza con n, t, m , donde n es la capacidad del transbordador, t es el tiempo que demora en ir de una orilla a la otra, y m es la cantidad de vehículos que debe procesar en este caso. Para las m líneas siguientes, se indica la hora de llegada de un automóvil (en minutos desde el comienzo del día), y la orilla a la que llega ('izquierda' o 'derecha').

Formato de salida

Para cada caso de prueba, genere una línea por automóvil, en el mismo orden que la entrada, indicando la hora en que ese automóvil se descarga en la orilla opuesta. Muestra una línea vacía entre casos. Puede suponer que $0 < n, t, m \leq 10000$. Los tiempos de llegada para cada caso de prueba son estrictamente no decrecientes. El ferry está inicialmente en la orilla izquierda. El tiempo de carga y descarga puede considerarse 0.

La salida se debe desplegar en la salida estandar (pantalla)

Ejemplo de entrada

```
2
2 10 10
0 left
10 left
20 left
30 left
40 left
50 left
60 left
```

70 left
80 left
90 left
2 10 3
10 right
25 left
40 left

Ejemplo de salida

10
30
30
50
50
70
70
90
90
110

30
40
60