



**Universidade de São Paulo**  
Instituto de Ciências Matemáticas e de Computação  
Departamento de Ciências de Computação  
SCC0202 e SCC0502 – Algoritmos e Estruturas de Dados I

## Trabalho 5 - Outras Listas

**Professor:** Dr. Marcelo Garcia Manzato (mmanzato@icmc.usp.br)  
**Estagiários PAE:** André (andrezanon@usp.br) e Luan (luanssouza@usp.br)

### Lista Backward

#### Descrição

Este trabalho consiste em implementar uma lista que vamos chamar de lista *backward*. Cada nó  $i$  na lista armazena, pelo menos, um apontador chamado *prox* para o próximo elemento na lista; um apontador chamado *back* para algum nó  $j$  na lista que está antes de  $i$  ou nulo; um inteiro chamado *chave*; e um inteiro chamado tempo que registra o momento discreto de tempo em que a chave foi inserida, contado a partir do tempo 0.

A inserção é feita sempre no fim da lista. Cada operação de inserção ou remoção na lista incrementa o tempo em uma unidade. As operações de inserção e remoção da lista devem ser implementadas como subrotinas.

#### Entradas

Seu programa deve ler um conjunto de valores da entrada padrão. A entrada contém um conjunto de linhas. Cada linha tem três formas possíveis:

- O caractere  $i$  seguido de dois inteiros  $n$  e  $j$ , significando que um novo nó deve ser inserido na lista com campo chave igual a  $n$  e *back* referenciando o nó  $j$  posições antes do nó que será inserido. Se  $j = 0$ , o apontador *back* deve ser nulo.
- O caractere  $r$  seguido de um inteiro  $n$ , significando que o primeiro nó que contenha a chave  $n$  deve ser removido e que todos os nós com apontadores *back* referenciando esse que será removido devem ser atualizados para nulo.
- O caractere  $f$  indicando o fim da entrada.

#### Saídas

Seu programa deve imprimir uma linha, que é o resultado da impressão das chaves e do tempo armazenado nos nós da lista ao percorrê-la através dos apontadores *prox*, no formato que aparece no exemplo abaixo. Quando um nó contiver um apontador *back* não-nulo, o programa deve imprimir, após o tempo, a posição do nó apontado na lista, sendo que o primeiro elemento da lista está na posição 0. Se a lista estiver vazia, o programa deve imprimir -1.

#### Observações

- O uso de listas é obrigatório.
- Somente as bibliotecas *stdio.h*, *stdlib.h* e *string.h* podem ser utilizadas.

# Exemplo

Na Figura 1, é mostrado a lista resultante das entradas do exemplo. Cada nó possui 4 "partições" em que a primeira é o valor do nó, a segunda é o valor do tempo, a terceira é o ponteiro *back* e a quarta o ponteiro *prox*.

## Entradas

i 10 0  
i 3 1  
i 7 0  
i 15 2  
i 2 4  
r 3  
i 9 1  
f

## Saídas

[10,0] [7,2] [15,3] [2,4,0] [9,6,3]

Figura 1 – Ilustração da Lista Resultante

