

Exercício 05 – Embaralhar

Contexto

O Freddy, eterno esqueleto do ICMC, comprou um baralho na papelaria do CAASO e para impressionar os seus amigos quer aprender a embaralhar. O único problema é que ele só conhece pilhas e filas! Para embaralhar será necessário remover e inserir no início e no fim do baralho, por causa disso ele precisa de sua ajuda. O seu papel é construir um deque para ajudar Freddy a embaralhar, e assim impressionar toda a sociedade esquelética do ICMC.

Descrição

Construa a implementação da estrutura de dados DEQUE, utilizando a interface fornecida abaixo, implementando todas as funções descritas.

O programa irá usar dois deque, enumerados como 1 e 2, para embaralhar um baralho convencional. O baralho irá começar no deque 1 e a entrada irá instruir as trocas entre ambos os deque, no fim o deque 1 será colocado acima do deque 2. São impressos o baralho no início e no final do programa, além de todas as trocas que ocorrerem. Dessa forma, o programa é apenas usuário do seu TAD DEQUE e serve como maneira de checar a estrutura.

```
(Deque.h)
#ifndef DEQUE_H
#define DEQUE_H

typedef struct deque_ DEQUE;

DEQUE* deque_criar();
void deque_apagar(DEQUE** deque);
bool deque_inserir_inicio(DEQUE* deque, ITEM* item);
bool deque_inserir_fim(DEQUE* deque, ITEM* item);
ITEM* deque_remover_inicio(DEQUE* deque);
ITEM* deque_remover_fim(DEQUE* deque);
ITEM* deque_primeiro(DEQUE* deque);
ITEM* deque_ultimo(DEQUE* deque);
int deque_contar(DEQUE* deque);
bool deque_vazia(DEQUE* deque);
bool deque_cheia(DEQUE* deque);

#endif
```

Entrada

A entrada é iniciada pelo número N de trocas. A seguir serão inseridas N linhas, cada linha com a informação de qual deque a carta será removida, a posição da remoção, e a posição que a carta deve ser inserida no outro deque.

Exemplo:

Exemplo de Entrada
5 1 fim fim 2 fim inicio 1 fim inicio 1 fim inicio 2 fim fim

Saída

A saída será o deque no início da execução, a seguir as trocas que são realizadas e enfim o deque resultante.

Exemplo de Saída
O exemplo de saída está na descrição do exercício no runcodes.

Observações:

- O exercício deve ser desenvolvido individualmente por cada aluno, sendo este responsável por decidir as melhores opções de implementação.
- Somente as bibliotecas `stdio.h`, `stdlib.h` e `stdbool.h` podem ser utilizadas.
- O arquivo que será submetido ao RunCodes deve ser `.zip` apenas com a solução `Deque.c`.