

## INTRODUÇÃO À CIÊNCIA DA COMPUTAÇÃO I SCC0221

---

### Livro Corrompido

---

## 1 Introdução

Você, como um bom estudante, estava lendo um livro digital em seu computador para aprofundar seus conhecimentos, mas foi interrompido por uma brusca queda de energia. Ao ligar a sua máquina novamente, porém, notou que o arquivo do livro havia sido corrompido e novas linhas introduzidas a ele.

Por sorte, um programa de recuperação de arquivos corrompidos foi capaz de identificar qual o número de linhas válidas e quais são elas. Resta, então, que você desenvolva um programa em C que leia todas as linhas do livro corrompido e, dadas essas informações, imprima na tela somente as linhas válidas.

## 2 Requisitos

Para a leitura das linhas, o seu programa deverá implementar uma função `char *read_line()` que lê uma string, caractere por caractere, da entrada padrão, alocando dinamicamente memória para armazenamento, até encontrar um `'\n'` e retornar o resultado.

- O `'\n'` não deve estar incluso no retorno, ou seja, a entrada `"Exemplo\n"` deve resultar em `"Exemplo\0"`.
- Não é permitido assumir um tamanho máximo para a string que será lida.
- É permitido alocar memória em blocos maiores durante a leitura, no entanto, antes de ser retornada, a string deve ter seu tamanho redefinido para conter somente seus caracteres e o `'\0'`. A linha `"Exemplo\0"` terá 8 bytes.
- Não se preocupe com entradas que terminem em `'\r'` ou `"\r\n"`, elas não estarão presentes nos casos de teste.

Toda a memória alocada dinamicamente deve ser devidamente liberada usando a função `free()` antes do final da execução. O programa `valgrind` pode ajudar na checagem de erros.

## 3 Entrada

O seu programa deverá ler da entrada padrão um número  $n > 0$  de linhas, seguido das  $n$  linhas. Depois, deverá ler o número  $0 < v \leq n$  de linhas válidas seguido dos  $v$  índices  $0 \leq i < n$  que as identificam.

- Pode-se utilizar um vetor estático para armazenar os ponteiros das strings.
- Lembre-se de consumir o `'\n'` deixado pelo `scanf()` após a leitura do tamanho  $n$ .

## 4 Saída

Seu programa deverá imprimir na saída padrão as  $v$  linhas válidas identificadas por cada índice  $i$  separadas por um `'\n'`.

## 5 Exemplos de entrada e saída

- Exemplo 1

Entrada

```
1 5
2 0 que faz um bom programador?
3 dasoid $$ de9wdwe 632r2
4 Um bom programador faz codigos corretos, claro!
5 pas((*suiwef-711410 34014 134 ju%##&w
6 kym-y8 6*(0y56 1 24 &%h50345 4958
7 2
8 0 2
```

Saída

```
1 0 que faz um bom programador?
2 Um bom programador faz codigos corretos, claro!
```

## 6 Bônus

Tente melhorar a sua função para que ela passe a receber um parâmetro `char delimitador` e seja capaz de ler a string até que tal caractere seja encontrado.

**Bom trabalho! :)**