

## Instituto de Computação - Unicamp

### MC102 - Algoritmos e Programação de Computadores

# Laboratório 20 - A hora do Contra-Ataque!

---

Prazo de entrega: **30/06/2017 23:59:59**

Peso: **2**

*Professor:* Eduardo C. Xavier

*Professor:* Guido Araújo

*Monitor:* Arthur Pratti Dadalto

*Monitor:* Cristina Cavalcante

*Monitor:* Klairton de Lima Brito

*Monitor:* Luís Felipe Mattos

*Monitor:* Paulo Finardi

*Monitor:* Paulo Lucas Rodrigues Lacerda

*Monitor:* Pedro Alves

*Monitor:* Renan Vilas Novas

*Monitor:* Vinicius de Novaes Guimarães Pereira

## Descrição

---

Alerta de transmissão:

"Saudações agente. Mais uma vez seus serviços são necessários. Estamos preparando um novo ataque sobre as bases do império no planeta XI35-ALPHA. O planeta XI35-ALPHA foi dividido em setores, e para cada setor criamos o seu respectivo mapa. Para otimizar o uso de nossas tropas, somente setores que apresentarem mais bases do império do que bases da aliança rebelde que receberão tropas auxiliares. A partir do mapa de cada setor você deve determinar qual deles irá receber tropas auxiliares. Como o planeta apresenta uma

quantidade gigantesca de setores, precisamos que isso seja feito da maneira mais rápida possível. Os dados da missão estão no arquivo criptografado que você acabou de receber.  
Que a força esteja com você!"

## Objetivo

---

Como engenheiro de software da aliança rebelde, você deve criar um programa que, dado um mapa de um setor seja capaz de determinar quantas bases do império e quantas bases rebeldes estão presentes naquela região.

Dica:

*A melhor forma de contar pode ser destruindo.*

## Entrada

---

A entrada consiste de dois números inteiros que representam respectivamente o número de linhas e colunas de uma matriz de caracteres, o número máximo de linhas e colunas será **100**. Em seguida, é dada a matriz de caracteres com a quantidade de linhas e colunas estabelecidas previamente. Na matriz, o caractere '\*' representa parte de uma base do império, o caractere '#' representa parte de uma base dos rebeldes e o caractere '-' representa uma área não construída.

Além disso, as bases não possuem uma forma geométrica definida. Duas bases de um mesmo grupo (império ou aliança) não podem estar encostadas uma na outra, i.e., não há duas posições vizinhas que pertençam a duas bases diferentes do império (ou da aliança). Dada uma posição  $(l, c)$  do mapa, definimos como sendo suas posições **vizinhas** aquelas que estão imediatamente acima  $(l-1, c)$ , abaixo  $(l+1, c)$ , à esquerda  $(l, c-1)$ , à direita  $(l, c+1)$ , nas diagonais superiores  $(l-1, c-1)$  e  $(l-1, c+1)$ , e nas diagonais inferiores  $(l+1, c-1)$  e  $(l+1, c+1)$ . Uma base é determinada pelo conjunto de caracteres # (ou \*) que são alcançáveis entre si. Dizemos que uma posição  $(l_0, c_0)$  é alcançável por uma posição  $(l_i, c_i)$  se existe um caminho entre  $(l_0, c_0)$  e  $(l_i, c_i)$  formada pelas posições  $(l_0, c_0), (l_1, c_1), (l_2, c_2), \dots, (l_i, c_i)$  (todas contendo o caractere #), tal que todo par de posições  $((l_{x-1}, c_{x-1}), (l_x, c_x))$ , para  $1 \leq x \leq i$ , são vizinhas.

Veja o exemplo de entrada abaixo.

```
5 5
- - # - -
- # # - -
# - - * *
- - * - -
- * - - -
```

No exemplo acima, temos uma base do império e uma base rebelde.

*Dica para a linguagem C:* leia a entrada da seguinte forma:

```
for(i = 0; i < rows; i++)
    for(j = 0; j < cols; j++)
        scanf("%c ", &map[i][j]);
```

## Saída

---

A saída é composta de duas linhas, a primeira deve informar a quantidade de bases rebeldes e a segunda deve informar a quantidade de bases do império. *Dica para a linguagem C:* imprima a saída da seguinte forma:

```
printf("Bases rebeldes: %d\n", rebels);
printf("Bases do imperio: %d\n", empire);
```

## Exemplos

---

### Teste 01

#### Entrada

```
10 10
- - - - - * - * * -
* - * * - - * * * -
* * * * * * * * * -
- * - - * # - - - *
# * * - * - * - - *
```

```

- - - * * - * - -
- * * * * - - * * -
- - - * - - * - * *
* - - - * - - * * *
* * * - - - - * * -

```

## Saída

Bases rebeldes: 2  
Bases do imperio: 2

## Teste 04

### Entrada

```

21 23
* - - * * - # - - * * - * * * - - - * - - -
- * * * - * * * * * * * - - - * * - - - * * -
- # * - * - * - * - # * - - * * * * * - * * -
* * * # - * - - * * * * - - - - * # - - * * #
* * * - * - - - * * - - # * - - - # # * - - -
* # * * * - * - * * - - * - - * - * * - - - *
* * * - # * * - * - - * * - - - * * # - * * -
* * * # - * - * * * - - * - * - * * * - * - -
# * - # - - - - * * * - - * - # - - * * * * *
* - - - * - * - - - - * * * * * * - * - * # -
* * - * * - - * - - - - # - - * - * - # - *
- - - - * - - * * - - * * * * - * * # * * - -
- - - - - * - - * * * * - * - * * * * - - * #
* - * * - * * * - - * - * - - * - - * - * - *
* - * - - * * - - * - * * * - - - * * * - - *
- * * * - - - * * * - - - - - - * - - - * - *
* * * * * - * - * * * * * - * - - - - - * #
* # - * - * * * - * * * - * - * * - * # - - -
- - * # - * * * * - * - # - - * * - - - - -
- * * * * - * - * - - * # * - - * - * - * - *
- * * * * - - - * - - - - * - - - * * - - -

```

## Saída

Bases rebeldes: 21  
Bases do imperio: 7

## Teste 12

### Entrada

45 45

- # # - # - \* # # # - \* \* # # - - # # \* # \* \* # # \* - # # # \* # #  
\* # # \* \* - \* - \* # # \* # # - # # # \* - \* \* - \* # # - \* \* # - # -  
- - - # - \* \* - \* - \* - # # # - # \* # - # # # # \* \* \* \* - \* \* # -  
\* # # # # - - # - # # \* \* # \* - # # - - # # - \* \* - - \* \* - - \* #  
- \* - - \* - \* - # # \* # - \* - \* - # # \* \* \* - # # \* # \* - # \* \* -  
- # # \* \* - - - # # \* - \* \* \* # # - \* # \* - \* # \* \* - \* \* \* \* # -  
# - \* # - \* - \* # \* - # \* # - \* # - # - # \* \* \* - \* \* # \* # # - #  
- # # \* \* - - \* \* \* \* # \* \* # - # # \* \* - - - \* \* \* # # \* # \* # \*  
# # \* \* - - \* \* # # \* - # # \* - # - # # \* \* \* - \* # \* - - # # \* \*  
# # # - \* # - # # # - \* - \* - # - \* \* # # # # - # - - \* - - \* - #  
# # - - \* - \* # # # - # # - # \* \* - # # # \* - # - \* # - # # \* - #  
- # # \* - - # # \* - \* \* # # - # # # # \* \* # \* - - # - - # # \* # \*  
# - # - \* \* - \* # # # \* # \* \* \* \* # # \* \* # # # # # # \* # \* # -  
\* \* - - # # # # # # \* \* # # - \* # - \* - - \* # # # # \* - \* \* # # \*  
# - \* \* # \* # # # # \* # - \* \* # - # # \* - # - # # # \* \* # # \* \* \*  
# # \* # # - \* \* \* - \* # \* # - # # # # # - \* - \* \* - - \* - - \* - #  
\* \* - \* - # # # # - # \* # \* # - \* \* # # # \* # \* \* - # \* # \* \* \* #  
# - # # \* # \* - # \* - # # # \* # # # - # # # # # - \* \* - # \* \* \* -  
# \* - - # # - # - # # - # \* \* # \* # # - - # # - - # # # # # \* # #  
# # # \* # # \* - \* - # \* - \* - # # - # \* # - - # # \* # # # \* - \* #  
# # \* \* # # # # # # # # # # - \* # # \* \* # # # # # - # - \* \* # - \*  
# # # # # \* # - - \* # - # - # \* \* \* \* # \* - \* - # - \* # - - \* # -  
# # \* # - - # \* # \* \* \* # \* \* - - # \* # - - \* # - \* \* # \* \* \* # #  
- - # \* # # # # # # \* \* # - # \* # # \* # - - # \* \* \* \* - # - - # \*  
# \* \* # # \* - \* # \* - - # \* \* - # \* \* \* # \* \* - - # # # # \* \* # \*  
- # # # - # \* # # # \* \* \* \* \* # - \* \* - \* \* \* \* \* \* # # \* \* # # -  
\* # # \* \* \* # - \* # # # \* \* \* - \* \* - \* - # # - \* \* \* # - # \* \* \*  
# - \* \* \* \* \* - - \* - \* - - # \* - - # - # \* # - # \* # - - \* # # -  
\* # \* \* # - \* \* # # # - # - # # # # # # - # - # \* \* \* # # \* \* # #  
# \* \* # \* # # # \* \* \* # \* # \* - \* - # # # - # # # # - # # - \* \* #  
- # \* # # # \* \* - # - - \* # \* - \* \* # # # \* - \* - \* # - \* - # # \*  
# - \* \* # # - # # # \* \* - # # # - \* # # \* # # \* - # - # # - \* \* \*  
# # - # # # \* \* \* - # \* \* # \* # # # - # - \* - \* - # - # \* \* - \* \*  
# - \* - # - # \* # - # \* # # # \* \* - - # \* - \* # # - # - # # \* - \*  
\* # \* # \* - \* \* # \* - # \* \* - # # # \* \* # - \* # \* # # - # - \* # \*  
# \* - \* # - \* - # # # # - - # \* \* - \* - # \* # \* - \* # # - # - - \*  
# # - \* # \* # # # \* - - # # # # - \* \* \* # # # # # - # # # - # # #  
- # # \* # # - \* \* - \* # - - # # \* - # - # # # \* # - # # # # # #  
# \* \* - - # # \* # - # \* - # - \* \* # # - # - # # # \* - - \* # # \* #  
# - - # # # \* \* \* - \* \* # # \* # # \* # \* # - \* - # - - \* - \* \* \*

```
# * * - - * * * - # * # # * * # # - # * # # # - # - * # * # # # #
* * * # * # # # - # - * * - - * # * * # # - # # # - * # * # # # *
* # # - # * * # * # * # # * - - # * * - * # * # # - # * # - - # #
# # # * - # # # # # * - * - * * # # # * # # * * * * - # * # - # -
```

## Saída

Bases rebeldes: 30

Bases do imperio: 81

Para mais exemplos, consulte os [testes abertos no Susy](#).

## Observações

---

- O número máximo de submissões é **15**;
- O seu programa deve estar completamente contido em um único arquivo denominado `lab20.c`.
- Para a realização dos testes do SuSy, a compilação dos programas desenvolvidos em C irá considerar o comando:  

```
gcc -std=c99 -pedantic -Wall -o lab20 lab20.c ;
```
- Você deve incluir, no início do seu programa, uma breve descrição dos objetivos do programa, da entrada e da saída, além do seu nome e do seu RA;
- Indente corretamente o seu código e inclua comentários no decorrer do seu programa.

## CrITÉrios importantes

---

Independentemente dos resultados dos testes do SuSy, o não cumprimento dos critérios abaixo implicará em nota zero nesta tarefa de laboratório.

- Para os programas em linguagem C, os únicos headers aceitos para essa tarefa são o `stdio.h` e `stdlib.h`.