# Lista 4 - Laços de repetição aninhados

**Prazo de entrega:** 15/03/2024 até 23:59

**Instruções de entrega**: Você deve ter um repositório em seu github para a disciplina de Programação procedural chamado **Programacao-Procedural**.

Crie uma pasta dentro dele chamada Listas e dentro dela crie uma pasta chamada Lista4. Para cada exercício da lista deve ser criado um projeto dotnet.

Ao terminar a lista, suba seus exercícios no github e envie o link no formulário abaixo.

**Link para o formulário de envio**: <https://forms.gle/FQv9nzpGbNJ2ucp57>

**MONITORIAS**: Serão realizados quatro dias de monitoriaspara auxiliar com a resolução dos exercícios. As datas e horários das monitorias são:

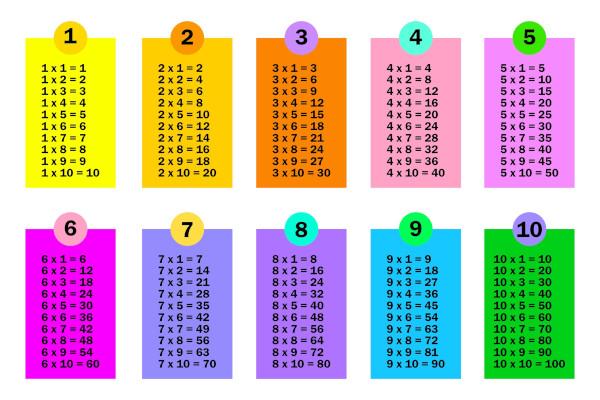
| **Dia** | **Horários** | | |
| --- | --- | --- | --- |
| Segunda-Feira | 19h até 21h |  |  |
| Terça-Feira | 19h até 21h | - | - |
| Quarta-Feira | 09h até 11h | 19h até 21h | - |
| Quinta-Feira | 19h até 21h | - | - |
| Sexta-Feira | 09h até 11h | 19h até 21h | 22h até 23:59 |

**OBS**: Se você não conseguir participar das monitorias nos dias e horários propostos, você pode solicitar uma monitoria em uma data e horário diferente, desde que seja comunicado com pelo menos um dia de antecedência 🙂

**Se você quiser** verificar se seu código está correto, você pode acessar o nosso [contest no Hacker rank](https://www.hackerrank.com/lista-4-lacos-de-repeticao-aninhados)! Ele te permitirá testar seu código e verificar se o resultado produzido está correto. Basta acessar o link e clicar no botão Sign up.

(Você não é obrigado a fazer isso, é apenas uma forma de validar se seu código está correto)

## **Exercícios**

1. Você está trabalhando em uma plataforma de aulas como desenvolvedor(a) de software. Sua equipe fez uma reunião com o cliente e o mesmo disse que sentiu falta de algumas coisas na plataforma. Nas palavras dele: “Poderiamos colocar a tabuada da multiplicação nas aulas de matemática, acho que isso ajudaria os alunos a memorizar os resultados”.  
   

Após essa reunião, sua líder técnica te pediu para criar um programa que utilize laços de repetição aninhados para escrever a tabuada da multiplicação, de maneira similar à imagem acima, porém **uma embaixo da outra**.

Crie um programa que exiba todas as tabuadas do 1 até o 10, uma embaixo da outra

**⚠️ATENÇÃO**: Após imprimir uma tabuada completa, você deve pular uma linha antes de começar a imprimir a próxima⚠️

**Exemplo de entrada**:

* não há entrada

**Saída esperada**:

*1 x 1 = 1*

*1 x 2 = 2*

*1 x 3 = 3*

*.*

*.*

*.*

*1 x 10 = 10*

*→ pulo de linha*

*2 x 1 = 2*

*2 x 2 = 4*

*2 x 3 = 6*

*.*

*.*

*.*

*2 x 10 = 20*

*3 x 1 = 3*

*3 x 2 = 6*

*3 x 3 = 9*

*.*

*.*

*.*

*3 x 10 = 30*

*→ pulo de linha*

*4 x 1 = 4*

*4 x 2 = 8*

*4 x 3 = 12*

*.*

*.*

*.*

*4 x 10 = 40*

*→ pulo de linha*

*5 x 1 = 5*

*5 x 2 = 10*

*5 x 3 = 15*

*.*

*.*

*.*

*5 x 10 = 50*

*→ pulo de linha*

*6 x 1 = 6*

*6 x 2 = 12*

*6 x 3 = 18*

*.*

*.*

*.*

*6 x 10 = 60*

*→ demais tabuadas…*

*10 x 1 = 10*

*10 x 1 = 20*

*10 x 1 = 30*

*.*

*.*

*.*

*10 x 10 = 100*

1. Você foi recém contratado por uma empresa para a vaga de programador de jogos digitais e seu supervisor decidiu testar seus conhecimentos em programação. Para isso, ele te passou a tarefa de criar um algoritmo que imprima na tela um triângulo retângulo, cuja altura será determinada pelo usuário.

Você deve, portanto, criar um programa que recebe um número inteiro e imprime na tela um triângulo retângulo contendo a quantidade de linhas passada pelo usuário, mas calma, existem algumas regras:

* Se a altura digitada pelo usuário for menor que um ou maior que nove, seu programa deve exibir a mensagem “altura invalida, insira uma altura entre 1 e 9” e receber novamente o valor da altura (deve fazer isso infinitas vezes, até que o usuário digite um valor entre 1 e 9).
* Caso a altura digitada pelo usuário seja válida, seu programa deve imprimir o triângulo exibindo os números em ordem crescente, conforme demonstrado abaixo.

**Exemplo de entrada**:

* 7

**Saída esperada**:

1

12

123

1234

12345

123456

1234567

**Outro exemplo de entrada**:

* 12 → o usuário tentou digitar uma altura inválida
* 5

**Outro exemplo de saída**:

altura invalida, insira uma altura entre 1 e 9

1

12

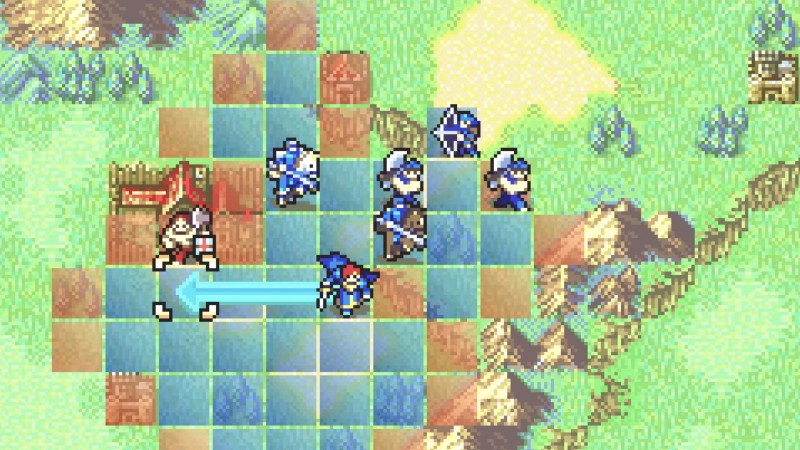
123

1234

12345

1. ENCURRALAR!!

Você está trabalhando em uma empresa que desenvolve jogos digitais e foi alocado para ajudar no desenvolvimento de um remake de um clássico jogo de game boy, ***Fire Emblem!***



Fire emblem é um jogo similar ao Xadrez, em que você precisa posicionar os membros de seu exército nas casas que deseja, de maneira estratégica.

Seu líder técnico te disse que você deve implementar a funcionalidade “encurralar”. Nas palavras dele: “***Nós temos um tabuleiro 8x8 e devemos criar um programa que preencha as casas que estão na extremidade do tabuleiro com a letra “s”. Todas as casas que não estão na extremidade devem permanecer vazias***”.

⚠️ [Link para a classe tabuleiro](https://github.com/MateusDebut/Programacao-Procedural/blob/main/Aulas/aula8/Tabuleiro.cs) → copie a classe e cole no seu repositório, como fizemos na aula 8

⚠️ Lembre-se de usar a função **Tabuleiro.InicializaTabuleiro()** antes de inserir as letras no tabuleiro

**Exemplo de entrada**:

* não há entrada

**Saída esperada**:

