

Projeto - Ligando os Canos

Importante:

- TODOS os membros do grupo devem participar e compreender completamente a implementação.
- Em caso de plágio, fraude ou tentativa de burlar o sistema será aplicado nota 0 na disciplina aos envolvidos.
- Alguns alunos podem ser solicitados para explicar com detalhes a implementação.
- Passar em todos os testes não é garantia de tirar a nota máxima. Sua nota ainda depende do cumprimento das especificações do trabalho, qualidade do código, clareza dos comentários, boas práticas de programação e entendimento da matéria demonstrada em possível reunião.

Esse projeto deverá ser realizado em grupo, preferencialmente com os grupos já definidos na disciplina. Este trabalho deverá ser implementado em linguagem C, C++ ou python3.

Nesse projeto iremos desenvolver um resolvidor para uma versão do jogo Flow Free. Nesse jogo temos um tabuleiro quadrado $n \times n$, em que $2k$ destas células estão previamente ocupadas com bolas coloridas, sempre 2 de cada cor. O objetivo do jogo é conectar as bolas de mesma cor sem que seus caminhos se cruzem. Cada célula do tabuleiro só pode conter 1 caminho de 1 cor.

Na Figura 1 (a) vemos um exemplo de um tabuleiro 5×5 com 8 bolas coloridas (2 amarelas, 2 verdes, 2 azuis e 2 vermelhas). Na Figura 1 (b) encontramos uma solução para essa instância do problema.

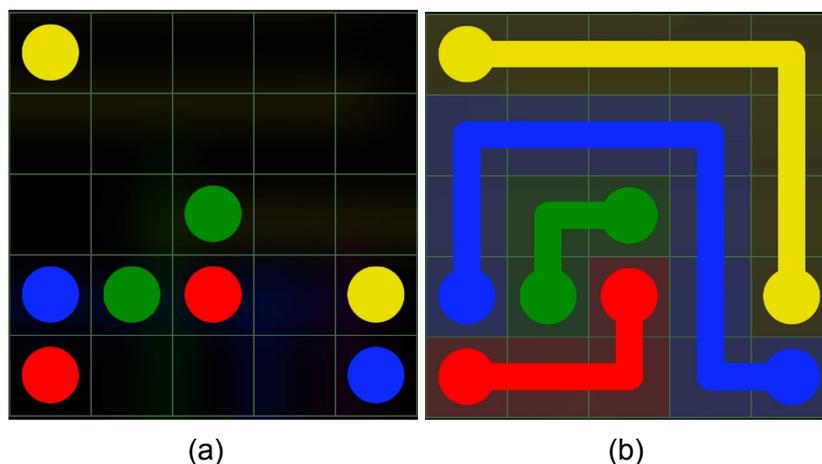


Figura 1: Exemplo do jogo Flow Free

No jogo original, ao final da resolução, por todas as células passa um caminho, na versão que iremos desenvolver isso não precisa acontecer, ou seja, podem ficar células sem nenhum caminho, porém todos os pares de bolas precisam estar conectados. Além disso, você também precisará elaborar uma instância que será “atacada” pelo seu algoritmo e pelos outros grupos.

Pela forma como será dada a saída, na nossa versão problema também será necessário evitar caminhos ambíguos, isso porque quando em formato de matriz não é possível distinguir qual o caminho do cano. Na Figura 2 o caminho da esquerda está proibido, tendo que ser trocado pelo da direita, na representação de matrix isso é equivalente a dizer que nenhuma casa tem 3 ou mais vizinhos iguais à cor da mesma.

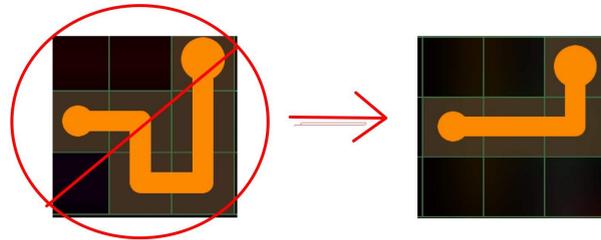


Figura 2: um caminho que fica ambíguo em formato de matrix, e deve ser proibido.

A entrada no nosso problema será constituída por um número inteiro n , depois seguido de n linhas, cada linha com n colunas, representando cada célula do tabuleiro. As cores serão representadas por letras maiúsculas que serão utilizadas na ordem do alfabeto.

(A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z),

Ou seja, se a instância utilizar 3 cores ela vai usar as letras A, B e C. Isso também limita qualquer instância a no máximo 26 cores distintas. As células vazias são representadas com 0. Abaixo temos um exemplo de uma instância com tabuleiro 5 x 5 e 3 cores.

```
5
0A000
0000A
000C0
0C0B0
B0000
```

Abaixo temos uma possível solução para o problema. Note que é possível que existam várias soluções diferentes.

```
5
0AAAA
0000A
0CCC0
0C0B0
BBBB0
```

Você precisará elaborar uma instância para o problema, que você irá enviar para que todos possam tentar resolvê-la. Obrigatoriamente o seu programa deve ser capaz de resolver sua instância em 10 segundos no sistema de execução [1]. O projeto terá um fator competitivo, o código de todos os grupos irão “enfrentar” todas as instâncias, e terão 10 segundos para essa resolução no sistema [1] usando uma única thread.

- Você pode olhar outros códigos da internet, porém não deve copiá-los. Se por acaso você usar um trecho de código, ou uma ideia de outro lugar, você deve compreender completamente esse código/ideia e deve dar os créditos.
- Você pode usar bibliotecas conhecidas e confiáveis elaboradas, com a condição de que você explique com antecedência como é feita a instalação no Ubuntu 20.04.2 LTS.
- No run.codes tem 3 casos de teste nas 3 linguagens aceitas. Se por ventura você quiser usar bibliotecas não suportadas pelo run.codes, fale com o professor que você poderá submeter de outra forma.
- Se você não tiver certeza se alguma coisa é permitida ou não no trabalho, não hesite em perguntar ao professor!

[1] Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz, usando apenas 1 thread, 10 segundos de execução e no máximo 20 GB de memória RAM, Ubuntu 20.04.2 LTS.

Datas de Entrega:

Instância: 25/07 pelo link <https://forms.gle/5Dpj3A3ZJvVFpoZ59>

Primeira Versão do Código: 02/08 via run.codes (ou a combinar)

Chance de resubmeter o Código: 04/08 via run.codes (ou a combinar)

Você também pode entregar um relatório de no máximo 2 páginas se achar que seu código demanda mais explicações, esse relatório não é obrigatório (por email hokama@unifei.edu.br)