

Aluno: André Santos Rocha

Ra: 235887

Problema 1: A - Love triangle

O problema nos fornece uma lista de n aviões na qual o i -ésimo valor $f[i]$ representa que o avião i "gosta" do avião $f[i]$. Nosso objetivo é encontrar um triângulo amoroso, ou seja, uma situação em que um avião A goste de B , B goste de C e C goste de A .

1.1 Ideia de solução

Para esse problema, percorremos a lista fornecida, de modo que, para cada avião, checamos a existência de um triângulo amoroso. Essa checagem ocorre da seguinte forma:

1. Se estamos na posição i da lista, definimos os aviões A, B e C como $A = i$, $B = f[i]$, $C = f[B]$.
2. Daí, verificamos $f[C]$. Se $f[C]$ for igual a A , encontramos um triângulo amoroso, imprimimos "YES" na tela e podemos parar a iteração.
3. Se chegarmos ao fim da iteração sem encontrarmos nenhum triângulo, imprimimos "NO" na tela.

1.2 Detalhes de implementação

Apesar da lista ser numerada de 0 a $n-1$, os aviões estão numerados de 1 a n . Sendo assim, podemos inicializar a posição 0 com um valor -1 (que nunca será uma entrada possível) e preencher a lista a partir da posição 1 até a posição $n+1$. Assim, evitamos erros devido ao manuseio incorreto dos índices.

Problema 2: C - Books Exchange

O problema nos fornece uma lista de n crianças na qual o i -ésimo valor $f[i]$ representa que a criança i emprestou seu livro para a criança $f[i]$ ao final de um dia. Nosso objetivo é encontrar o dia em que o livro retorna ao seu primeiro dono, para cada uma das crianças.

1.1 Ideia de solução

Para esse problema, percorremos a lista fornecida, de modo que, para cada criança, checamos o dia do retorno do livro para ela mesma. Essa checagem ocorre da seguinte forma:

1. Se estamos na posição i da lista, definimos a dona do livro $D = i$, a emprestadora $E = i$ e o dia $= 1$.
2. Daí, enquanto o livro não retornar a D , definimos a criança tomadora $T = f[i]$ e verificamos se T é igual a D . Se $T = D$, o livro retornou à sua dona, então imprimimos o dia. Se $T \neq D$, a emprestadora passa a ser $E = T$ e somamos 1 na variável dia.

1.2 Detalhes de implementação

Apesar da lista ser numerada de 0 a $n-1$, as crianças estão numeradas de 1 a n . Sendo assim, podemos inicializar a posição 0 com um valor -1 (que nunca será uma entrada possível) e preencher a lista a partir da posição 1 até a posição $n+1$. Assim, evitamos erros devido ao manuseio incorreto dos índices.