

A interpretação da prova faz parte da avaliação, portanto leia com atenção.

Sejam duas listas de números inteiros, ambas ordenados e sem valores repetidos (considerando cada lista). O problema em questão é o de verificar, para cada valor da primeira lista, quais também existem na segunda lista, mantendo registro de suas posições para posterior processamento.

Considere a seguinte solução para esse problema:

```
unsigned char* funcao(unsigned short* v1, unsigned short* v2, int n) {
    unsigned char* found = calloc(n, sizeof(unsigned char));
    for (int i = 0; i < n; i++) {
        unsigned short value = v1[i];
        for (int j = 0; j < n; j++) {
            if (value == v2[j]) found[i] = 1;
        }
    }
    return found;
}
```

1. (3 pontos) Formule **a equação de recorrência** que representa o número de comparações e operações de acesso a vetor realizadas no pior caso, excluindo da contagem operações relativas a linhas com instruções **for** e **while**.

Considere a variável a como sendo acessos a vetor, e c as comparações.

Dicas: você pode dividir o algoritmo em partes para a contagem e depois somar. Para formular a recorrência, tente transformar o **for** mais externo em uma recursão.

A seguir, resolva a equação de recorrência, obtendo a **forma fechada** em termos de n e suas constantes a e c . É preciso mostrar seu raciocínio, não é suficiente apenas a equação final.

2. (4 pontos) **Otimize a solução acima, da melhor forma que você conseguir**, e submeta sua solução para o run.codes na **Avaliação [M-Z]**, assumindo:

- Vetores com 2 bytes sem sinal, alocados dinamicamente, na memória heap;
- Entrada: número total de elementos das listas (n)
- Método de geração das listas 1 e 2, copiar o código abaixo (exatamente igual, para que seja possível ter os mesmos valores que os esperados no run.codes

```
srand(1);
v1[0] = rand()%2;
for (int i = 1; i < n; i++) {
    v1[i] = v1[i-1]+(rand()%3)+1;
}
v2[0] = rand()%2;
for (int i = 1; i < n; i++) {
    v2[i] = v2[i-1]+(rand()%3)+1;
}
```

- Saída: elementos em comum às duas listas, em ordem, impressos no formato "%04hu ", finalizando com uma quebra de linha, "\n"
3. (3 pontos) **Conte as operações da sua solução**, utilizando contagem direta ou equação de recorrência. Para isso, escreva a formulação/derivação completa da sua análise, dando a **forma fechada da função de eficiência** em termos de n e suas constantes a e c .

Atenção: envie sua solução para as questões 1 e 3 no e-disciplinas, na Avaliação do Módulo 1, em formato imagem ou PDF, devidamente identificada com seu nome. Pode ser uma foto de uma folha ou a solução escaneada/digitalizada. Por favor, escreva de forma legível, use caneta com cor escura e boa iluminação - não será corrigido se eu não conseguir ler.

O envio dos arquivos em ambas as plataformas deve ter conteúdo devidamente identificado com seu nome (com número USP e nome), com desconto na nota caso não contenha identificação.