

## Prova 2 – 2024/2

### 1 Descrição

1. O único sistema operacional permitido para prova é o da maratona.
2. Usar funções em pelo menos 2 códigos, com passagem de parâmetro por valor e por referência. Também usar alocação dinâmica em pelo menos 1 código. O não uso de funções e alocação dinâmica acarretará em um decremento de 3 pontos na prova;
3. prova em [sharif.inf.ufg.br/hebert](http://sharif.inf.ufg.br/hebert)
4. Cada máquina do laboratório tem um endereço IP que é gravado pelo sistema, alunos que tenham logado com dois endereços IPs diferentes receberam nota zero. (a não ser em caso de troca de computador devidamente avisada);
5. Não será permitido o uso de celulares durante a prova;
6. A nota máxima considerada nesta prova é de 100 pontos;
7. Considerando uma nota de 0 a 10 na prova, a nota será o valor obtido pela soma dos exercícios corretos submetidos dividido por 10;
8. Cópias de código ou de partes de código de soluções implicarão em nota ZERO na questão correspondente para todos os envolvidos;
9. Utilize o sharif como seu backup, não confie na máquina pois a mesma pode travar, assim submeta quantas versões quiser para cada código.

## Exercícios

### 1. Número Primo na Matriz (35 pontos)

O aluno Mario juntamente com a aluna Dani perceberam que uma tarefa muito fácil poderia cair na prova de Álgebra Linear e resolveram tentar respondê-la. A tarefa é encontrar o primeiro número primo “ $P$ ” diferente de 1 e 2 na diagonal principal de uma matriz “ $A$ ” de dimensão  $N \times N$ , em seguida multiplicar os elementos de cada coluna com paridade diferente da coluna onde  $P$  se encontra pelo valor de  $P$ . A ordem dos elementos na diagonal principal é definida pelos pares ordenados (1,1), (2,2), (3,3), e assim por diante. Caso não exista elemento primo diferente de 1 e 2 na diagonal principal da matriz  $A$ , pegar o último elemento da diagonal principal e multiplicar por cada elemento das linhas ímpares da matriz  $A$ . Considerar uma linha par ou ímpar pela posição em linguagem C, ou seja, linha zero é par, linha 1 é ímpar e assim por diante. Ao final mostrar o valor da matriz  $A$  após realizada as operações. O seu programa não deverá usar variáveis globais.

#### Entrada

A primeira linha contém um inteiro  $N < 501$ , indicando a dimensão da matriz  $A$ . As  $N$  linhas seguintes contém os elementos da matriz  $A$ .

#### Saída

Seu programa deve produzir  $N$  linhas e  $N$  colunas com o valor da matriz  $A$  alterado.

#### Exemplos

Entrada 1	Saída 1
5	
1 2 3 4 5	3 2 9 4 15
2 3 4 5 6	6 3 12 5 18
7 8 9 10 11	21 8 27 10 33
8 9 10 11 12	24 9 30 11 36
9 10 11 12 13	27 10 33 12 39

Entrada 2	Saída 2
3	
1 2 3	1 2 3
2 1 4	8 4 16
5 3 4	5 3 4

## 2. Nomes (35 pontos)

A garota cimairam trabalha com nomes todo tempo e deseja saber se algum nome ao contrário forma um nome conhecido por ela. Você deverá fazer um programa que dado um conjunto de nomes escritos por cimairam, imprimir os nomes ao contrário e em ordem alfabética, a ordem alfabética deverá ser considerada com os nomes ao contrário. Não usar função de ordenação que já esteja pronta na linguagem de programação, por exemplo `qsort( )`. Considere que a ordem alfabética é a mesma dada pela tabela ASCII. Não usar variáveis globais.

### Entrada

A primeira linha possui dois inteiros  $N < 101$  e  $M < 501$ ,  $N$  é a quantidade de nomes que cimairam irá digitar e  $M$  é a quantidade máxima de letras que cada nome deverá ter.

### Saída

Seu programa deve produzir  $N$  linhas, contendo os nomes ao contrário em ordem alfabética. As letras nos nomes da saída tem letras minúsculas e maiúsculas conforme aparecem na entrada.

### Exemplos

#### Entrada

```
4 10
Notielc
Nosila
Rutra
Ana
```

#### Saída

```
alisoN
anA
artuR
cleitoN
```

### 3. Estudar não doí (35 pontos)

Alban vive escutando da sua mãe que estudar não doí. Atualmente, Alban entrou na Universidade para cursar Ciência da Computação e esta tendo uma disciplina que é muito difícil, assim ele não esta concordando muito com sua mãe. Nesta disciplina Alban tem que fazer muitas manipulações com conjuntos. Devido ao tamanho dos conjuntos Alban passa horas para testar propriedades simples. Neste momento Alban esta verificando se elementos de dois conjuntos  $A$  e  $B$  são pares ou ímpares. Sua tarefa é ajudar Alban a encontrar os elementos pares do conjunto  $A$  e encontrar os elementos ímpares do conjunto  $B$  e formar um novo conjunto  $C$  com tais elementos. O conjunto  $C$  deverá conter a concatenação entre os pares de  $A$  e ímpares de  $B$ , ou seja, um elemento par do conjunto  $A$ , depois um elemento ímpar do conjunto  $B$ , depois o próximo elemento par do conjunto  $A$  e assim por diante. Se os elementos do conjunto  $A$  terminarem primeiro que os elementos do conjunto  $B$ , então as últimas posições de  $C$  só terão elementos ímpares de  $B$ . Caso os elementos de  $B$  terminem antes dos elementos de  $A$ , então as últimas posições de  $C$  só terão elementos pares de  $A$ .

#### Entrada

A primeira linha possui dois inteiros  $N < 5001$  e  $M < 5001$ ,  $N$  é a quantidade de elementos do conjunto  $A$  e  $M$  é a quantidade de elementos do conjunto  $B$ . A segunda linha contém os elementos do conjunto  $A$ . A terceira linha contém os elementos do conjunto  $B$ .

#### Saída

Seu programa deve produzir o conjunto  $C$ , com os elementos pares do conjunto  $A$  e os elementos ímpares do conjunto  $B$ . Após o último valor deverá haver um espaço.

#### Exemplos

##### Entrada 1

```
4 10
1 2 3 4
5 7 8 9 0 6 11 14 13 10
```

##### Saída 1

```
2 5 4 7 9 11 13
```

##### Entrada 2

```
5 3
6 1 3 11 13
5 7 9
```

##### Saída 2

```
6 5 7 9
```