



**3ª AVALIAÇÃO**  
**05/03/2022**

**ORIENTAÇÕES GERAIS:**

1. Siga atentamente as especificações da tarefa quanto ao formato da entrada e saída de seu programa.
2. Não implemente nenhum recurso gráfico nas suas soluções (janelas, menus, etc), nem utilize qualquer rotina para limpar a tela ou posicionar o cursor.
3. Todas as tarefas têm o mesmo valor na correção.
4. As tarefas não estão ordenadas por ordem de dificuldade.
5. Não utilize arquivos para entrada ou saída nos programas. Todos os dados devem ser lidos da entrada padrão (tipicamente o teclado) e escritos na saída padrão (tipicamente a tela).
6. Procure resolver o problema de maneira eficiente. Na correção, eficiência também será levada em conta. As soluções serão testadas com outras entradas além das apresentadas como exemplo nas tarefas.
7. A implementação pode ser feitas em grupos de ATÉ 3 (três) alunos.
8. No início do código fonte deve haver um comentário informando os nomes dos autores, sob pena de nulidade da questão.
9. Os autores de cada implementação poderão ser chamados para explicar o código entregue.

**ENTREGA:**

- Os códigos produzidos devem ser entregues em um único arquivo compactado em formato ZIP.
- O arquivo compactado deverá ser enviado para o e-mail: [eyder@phb.uespi.br](mailto:eyder@phb.uespi.br)
- O **assunto do e-mail** deverá iniciar com “PROG1 – AVAL3: ” seguido dos primeiros nomes dos autores.
- A entrega deverá ser feita até o final do dia 07/03/2022.
- Códigos considerados copiados da Internet ou de outras equipes receberão NOTA ZERO, independente de quem seja(m) o(s) autor(es).



### **PROBLEMA 1:**

Noel está pedindo sua ajuda para distribuir seus presentes de maneira ótima. Ele te deu uma lista contendo um número  $N$ , seguido por  $N$  inteiros  $g_i$ , e te contou que estas anotações eram sobre  $N$  crianças que ele andou observando durante este ano. Cada um destes  $N$  números representa quantas boas ações cada criança fez.

Agora ele quer distribuir presentes, mas ele quer ser justo em relação à quantas boas ações cada criança fez e quantos presentes elas merecem. Ele te deu três restrições:

1. Toda criança deve receber no mínimo 1 presente.
2. Para todo par de crianças  $A$  e  $B$ , tal que ambas fizeram a mesma quantidade de boas ações, ambas merecem receber a mesma quantidade de presentes.
3. Para todo par de crianças  $A$  e  $B$ , tal que a criança  $A$  fez mais boas ações que a criança  $B$ , a criança  $A$  merece receber mais presentes que a criança  $B$ .

Por exemplo, vamos supor que há 3 crianças, e que a primeira criança fez 1 ato bom este ano, a segunda fez 3 atos bons, e a terceira fez 1 ato bom. Uma forma válida de distribuir presentes seria dar 3 presentes para a primeira e terceira criança (pois elas fizeram a mesma quantidade de atos bons), e 5 presentes para a segunda criança (pois ela fez mais atos bons que as outras). Note que esta distribuição respeita as restrições, mas ela não é a única, e nem a mais econômica forma de distribuir os presentes.

Após receber as anotações de Noel, você deve ajudá-lo a descobrir a quantidade total mínima de presentes a serem enviados às crianças. Você deve lembrar de respeitar as restrições estabelecidas, e deve tentar minimizar quantos presentes serão enviados no total.

### **FORMATO DA ENTRADA**

Cada entrada possui vários casos de teste. A primeira linha de cada caso de teste contém um inteiro  $N$  ( $1 \leq N \leq 10^4$ ), representando a quantidade de crianças à quem os presentes serão distribuídos.

A segunda linha de entrada contém  $N$  inteiros  $g_i$  ( $1 \leq g_i \leq 10^4$ , para todo  $1 \leq i \leq N$ ), representando que a  $i$ -ésima criança fez  $g_i$  boas ações neste ano.

A entrada termina com um valor de  $N=0$ .

### **FORMATO DA SAÍDA**

Você deve imprimir uma linha, contendo um inteiro, representando a quantidade total mínima de presentes que devem ser enviados às crianças.

#### **Exemplo de entrada**

```
3
1 3 1
3
1 3 2
0
```

#### **Saída correspondente**

```
4
6
```



**UNIVERSIDADE ESTADUAL DO PIAUÍ - UESPI**  
**CURSO DE BACHARELADO EM COMPUTAÇÃO**  
**DISCIPLINA: PROGRAMAÇÃO I**  
**PROFESSOR: EYDER RIOS**

**PROBLEMA 2:**

Clodovildo Procariontes Urus é um Guru da Sorte, ele usa numerologia como uma ciência que estuda a sorte a partir de números. Um exemplo de aplicação da numerologia está no cálculo da sorte a partir do nome da pessoa. Muitos preocupados com a sorte acabam mudando o nome para garantir uma sorte melhor. O cálculo do número da sorte é baseado na tabela pitagórica, onde cada letra é traduzida para um número:

1	2	3	4	5	6	7	8	9
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	

Uma vez traduzida a letra por números, fazemos a soma, e somamos os dígitos dos resultados até sobrar um valor entre 1 e 9. Veja por exemplo o nome "Harry Potter":  $8+1+9+9+7+7+6+2+2+5+9 = 65 \rightarrow 6+5 = 11 \rightarrow 1+1 = 2$ . O número da sorte para "Harry Potter" é 2. Clodovildo solicitou a você um programa que calcule o número da sorte a partir de nomes fornecidos, para que ele possa declarar a sorte da pessoa.

**FORMATO DA ENTRADA**

A entrada contém vários casos de teste. Cada caso de teste apresenta um nome N em uma linha única, o nome é composto por letras sem acentuação, e o cedilha está representado como a letra 'C'. Cada nome tem no máximo 100 caracteres.

**FORMATO DA SAÍDA**

Para cada caso de teste da entrada seu programa deve gerar para nome de entrada, uma linha com o número da sorte.

**Exemplo de entrada**

**Saída correspondente**

Harry Potter  
Hermione Granger  
Albus Dumbledore

2  
4  
1



### **PROBLEMA 3:**

A família toda ficou excitada pela novidade. Todos sabiam que o meu avô tinha sido um excelente jogador de bridge por décadas, mas quando foi anunciado que ele estaria no Guinness Book, o livro dos recordes, como o jogador de bridge de maior sucesso de todos os tempos, wow, aquilo foi surpreendente.

A Associação Internacional de Bridge (AIB) tem mantido, por diversos anos, um ranking semanal dos melhores jogadores do mundo. Considerando que cada aparição em um ranking semanal constitui um ponto para o jogador, meu avô foi nominado o melhor jogador de todos os tempos porque ele conseguiu o maior número de pontos.

Tendo muitos amigos que também estavam competindo com ele, meu avô está extremamente curioso para saber que jogador(es) ficou(aram) com o segundo lugar. Ele precisa de um programa, o qual, dada uma lista com os ranking semanais, descubra que jogador(es) ficou(aram) com o segundo lugar, de acordo com o número de pontos.

### **FORMATO DA ENTRADA**

A entrada contém diversos casos de teste. Jogadores são identificados por inteiros de 1 a 10000. A primeira linha de um caso de teste contém dois inteiros  $N$  e  $M$ , indicando, respectivamente, o número de rankings disponíveis ( $2 \leq N \leq 500$ ) e o número de jogadores em cada ranking ( $2 \leq M \leq 500$ ). Cada uma das próximas  $N$  linhas contém a descrição de um ranking semanal. Cada descrição é composta por uma sequência de  $M$  inteiros, separados por um espaço em branco, identificando os jogadores que apareceram naquele ranking semanal. Você pode assumir que:

- em cada caso de teste há exatamente um melhor jogador e ao menos um segundo melhor jogador,
- cada ranking semanal consiste de  $M$  jogadores distintos.

O final da entrada é indicado por  $N = M = 0$ .

### **FORMATO DA SAÍDA**

Para cada caso de teste da entrada seu programa deve produzir uma linha de saída, contendo o identificador do jogador que é o segundo melhor, em número de aparições nos rankings. Se há um empate para segundo lugar, imprima os identificadores de todos os segundo colocados, em ordem crescente. Cada identificador produzido deve ser seguido por um espaço em branco.

#### **Exemplo de entrada**

```
4 5
20 33 25 32 99
32 86 99 25 10
20 99 10 33 86
19 33 74 99 32
3 6
2 34 67 36 79 93
100 38 21 76 91 85
32 23 85 31 88 1
0 0
```

#### **Saída correspondente**

```
32 33
1 2 21 23 31 32 34 36 38 67 76 79 88 91 93 100
```



**UNIVERSIDADE ESTADUAL DO PIAUÍ - UESPI**  
**CURSO DE BACHARELADO EM COMPUTAÇÃO**  
**DISCIPLINA: PROGRAMAÇÃO I**  
**PROFESSOR: EYDER RIOS**

**PROBLEMA 4:**

Brasil e Alemanha foi um jogo inesquecível. Neste jogo a Alemanha marcou 7 gols em 90 minutos, e o Brasil apenas 1. Por ser uma importante marca na história do futebol brasileiro, Leonardo resolveu fazer uma página web para que todos os brasileiros nunca se esquecessem desse dia magnífico.



A página mostrará qual o placar do jogo caso a partida ainda estivesse acontecendo desde aquele fatídico 8 de julho de 2014, considerando que a taxa de gols fosse mantida (Como mostra a figura abaixo). Para facilitar seu trabalho, Leonardo pede sua ajuda para calcular o placar que deverá ser mostrado na página web.

Como o objetivo do site é chocar o torcedor. Os gols sofridos pelo Brasil devem ser arredondado para cima enquanto os gols sofridos pela Alemanha devem ser arredondados para baixo.

**FORMATO DA ENTRADA**

A entrada é composta por um número  $N$  ( $90 \leq N \leq 10^9$ ) que indica quantos minutos se passaram desde o apito inicial da partida. A entrada termina quando  $N = 0$ .

**FORMATO DA SAÍDA**

A saída deve ser escrita em apenas uma linha: “Brasil B x Alemanha A” onde B e A são inteiros que indicam a quantidade de gols sofridos por Brasil e Alemanha respectivamente.

**Exemplo de entrada**

90  
100  
180  
0

**Saída correspondente**

Brasil 1 x Alemanha 7  
Brasil 1 x Alemanha 8  
Brasil 2 x Alemanha 14



### **PROBLEMA 5:**

Uma rainha requisitou os serviços de um monge e disse-lhe que pagaria qualquer preço. O monge, necessitando de alimentos, perguntou a rainha se o pagamento poderia ser feito em grãos de trigo dispostos em um tabuleiro de damas, de forma que o primeiro quadrado tivesse apenas um grão, e os quadrados subsequentes, o dobro do quadrado anterior.

A rainha considerou o pagamento barato e pediu que o serviço fosse executado, porém, um dos cavaleiros que estava presente e entendia um pouco de matemática alertou-a que seria impossível executar o pagamento, pois a quantidade de grão seria muito alta.

Curiosa, a rainha solicitou então a este cavaleiro que era bom em cálculo, que fizesse um programa que recebesse como entrada o número de quadrados a serem usados em um tabuleiro de damas e apresentasse a quantidade de kg de trigo correspondente, sabendo que cada 12 grãos do cereal correspondem a uma grama.

**Requisito:** o cálculo da quantidade deverá caber em um inteiro de 64 bits sem sinal.

### **FORMATO DA ENTRADA**

A primeira linha de entrada contém um único inteiro  $N$  ( $1 \leq N \leq 100$ ), indicando o número de casos de teste. Cada caso de teste contém um único inteiro  $X$  ( $1 \leq X \leq 64$ ), indicando o número de casas do tabuleiro que serão utilizadas.

### **FORMATO DA SAÍDA**

Para cada caso de teste, imprima a quantidade de kgs de trigo que o monge esperava receber.

#### **Exemplo de entrada**

3  
7  
19  
14

#### **Saída correspondente**

0 kg  
43 kg  
1 kg