

# OBI 2024 – Nível 2: Fase 1

*Concurso*

Prof. Edson Alves

*Faculdade UnB Gama*

Cláudia trabalha na OBI (Organização dos Bons Informáticos), que recentemente realizou um concurso para contratar novos funcionários. Agora, Cláudia tem a tarefa de determinar a *nota de corte* para o concurso. Chamamos de nota de corte a nota mínima necessária para ser aprovado no concurso. Ou seja, se a nota de corte do concurso for  $C$ , então todos os participantes com uma nota maior ou igual a  $C$  serão aprovados no concurso e todos com nota menor que  $C$  serão reprovados.

Seu chefe pediu para que Cláudia aprove no mínimo  $K$  candidatos do concurso para a próxima fase, mas ela também não quer que a nota de corte seja muito baixa. Por isso, Cláudia decidiu que a nota de corte deverá ser a maior nota  $C$  que faz com que no mínimo  $K$  candidatos sejam aprovados.

Sua tarefa é: dados o número  $N$  de candidatos, as notas  $A_1, A_2, \dots, A_N$  dos candidatos e a quantidade mínima de aprovados  $K$ , diga qual deve ser a maior nota de corte  $C$  para que pelo menos  $K$  candidatos sejam aprovados.

## Entrada

A primeira linha da entrada contém dois inteiros,  $N$  e  $K$ , representando, respectivamente, o número de participantes e o número mínimo de candidatos que devem ser aprovados.

A segunda linha da entrada contém  $N$  inteiros  $A_i$ , representando as notas dos participantes.

## Saída

Seu programa deve imprimir uma linha contendo um único inteiro  $C$ , a nota de corte que deve ser escolhida por Cláudia.

## Restrições

- ▶  $1 \leq K \leq N \leq 500$
- ▶  $1 \leq A_i \leq 100$  para todo  $1 \leq i \leq N$

## Informações sobre a pontuação

A tarefa vale 100 pontos. Estes pontos estão distribuídos em subtarefas, cada uma com suas **restrições adicionais** às definidas acima.

- ▶ **Subtarefa 1 (0 pontos):** Esta subtarefa é composta apenas pelos exemplos mostrados abaixo. Ela não vale pontos, serve apenas para que você verifique se o seu programa imprime o resultado correto para os exemplos.
- ▶ **Subtarefa 2 (20 pontos):**  $K = 1$ .
- ▶ **Subtarefa 3 (20 pontos):**  $K = 3$ .
- ▶ **Subtarefa 4 (20 pontos):**  $A_i \leq 2$ .
- ▶ **Subtarefa 5 (40 pontos):** Sem restrições adicionais.



## Exemplo de entrada e saída





## Exemplo de entrada e saída

3 1





## Exemplo de entrada e saída

3 1



*# de candidatos*





## Exemplo de entrada e saída

3 1  
↑  
*# de aprovados*







## Exemplo de entrada e saída

3 1  
92 83 98



## Exemplo de entrada e saída

3 1

92 83 98



*Nota do candidato 1*



## Exemplo de entrada e saída

3 1  
92 83 98



*Nota do candidato 2*



## Exemplo de entrada e saída

3 1

92 83 98



*Nota do candidato 3*

## Exemplo de entrada e saída

3 1  
92 83 98

92

83

98

## Exemplo de entrada e saída

3 1  
92 83 98

$$C = 70$$

92

83

98

## Exemplo de entrada e saída

3 1  
92 83 98

$$C = 70$$



## Exemplo de entrada e saída

3 1  
92 83 98

$$C = 99$$

92

83

98



## Exemplo de entrada e saída

3 1  
92 83 98

$C = 99$

92

83

98

## Exemplo de entrada e saída

3 1  
92 83 98

$$C = 90$$

92

83

98

## Exemplo de entrada e saída

3 1  
92 83 98

$$C = 90$$



## Exemplo de entrada e saída

3 1

92 83 98

$$C = 98$$

92

83

98

## Exemplo de entrada e saída

3 1  
92 83 98

$$C = 98$$



## Exemplo de entrada e saída

3 1  
92 83 98

↓  
98

$C = 98$



**Solução: Subtarefa 2 ( $K = 1$ )**

## Solução: Subtarefa 2 ( $K = 1$ )

- ★ Nesta subtarefa, pelo menos um candidato deve ser aprovado



## Solução: Subtarefa 2 ( $K = 1$ )

- ★ Nesta subtarefa, pelo menos um candidato deve ser aprovado
- ★ Seja  $M$  a maior nota obtida entre todos os candidatos

## Solução: Subtarefa 2 ( $K = 1$ )

- ★ Nesta subtarefa, pelo menos um candidato deve ser aprovado
- ★ Seja  $M$  a maior nota obtida entre todos os candidatos
- ★ Se a nota de corte for igual a  $M$ , ao menos o candidato que obteve  $M$  será aprovado

## Solução: Subtarefa 2 ( $K = 1$ )

- ★ Nesta subtarefa, pelo menos um candidato deve ser aprovado
- ★ Seja  $M$  a maior nota obtida entre todos os candidatos
- ★ Se a nota de corte for igual a  $M$ , ao menos o candidato que obteve  $M$  será aprovado
- ★ Se a nota de corte for maior que  $M$ , ninguém será aprovado

## Solução: Subtarefa 2 ( $K = 1$ )

- ★ Nesta subtarefa, pelo menos um candidato deve ser aprovado
- ★ Seja  $M$  a maior nota obtida entre todos os candidatos
- ★ Se a nota de corte for igual a  $M$ , ao menos o candidato que obteve  $M$  será aprovado
- ★ Se a nota de corte for maior que  $M$ , ninguém será aprovado
- ★ Portanto, para esta subtarefa a resposta é a maior nota obtida entre todos os candidatos

```
#include <stdio.h>
```

```
int main()  
{
```

```
    for (int a = 1; a <= N; ++a)  
        for (int b = a; b <= N; ++b)
```

```
#include <stdio.h>
```

X

```
int main()
```

```
{
```

```
    int N, S;
```

```
    scanf("%d %d", &N, &S);
```

```
    for (int a = 1; a <= N; ++a)
```

```
        for (int b = a; b <= N; ++b)
```

X

```
#include <stdio.h>
```

X

```
int main()
```

```
{
```

```
    int N, S;
```

```
    scanf("%d %d", &N, &S);
```

```
    int X[N + 1];
```

```
    for (int a = 1; a <= N; ++a)
```

```
        for (int b = a; b <= N; ++b)
```

X

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int N, S;
```

```
    scanf("%d %d", &N, &S);
```

```
    int X[N + 1];
```

```
    for (int i = 1; i <= N; ++i)
```

```
        scanf("%d", &X[i]);
```

```
        ps[i] = ps[i - 1] + X[i];
```

```
    for (int a = 1; a <= N; ++a)
```

```
        for (int b = a; b <= N; ++b)
```



```
#include <stdio.h>
```

X

```
int main()
```

```
{
```

```
    int N, S;
```

```
    scanf("%d %d", &N, &S);
```

```
    int X[N + 1];
```

```
    for (int i = 1; i <= N; ++i)
```

```
        scanf("%d", &X[i]);
```

```
    int ps[N + 1];
```

```
        ps[i] = ps[i - 1] + X[i];
```

```
    for (int a = 1; a <= N; ++a)
```

```
        for (int b = a; b <= N; ++b)
```

X

```
#include <stdio.h>
```

X

```
int main()
```

```
{
```

```
    int N, S;
```

```
    scanf("%d %d", &N, &S);
```

```
    int X[N + 1];
```

```
    for (int i = 1; i <= N; ++i)
```

```
        scanf("%d", &X[i]);
```

```
    int ps[N + 1];
```

```
    for (int i = 1; i <= N; ++i)
```

```
        ps[i] = ps[i - 1] + X[i];
```

```
    for (int a = 1; a <= N; ++a)
```

```
        for (int b = a; b <= N; ++b)
```

X

```
#include <stdio.h>
```

X

```
int main()
{
    int N, S;
    scanf("%d %d", &N, &S);

    int X[N + 1];

    for (int i = 1; i <= N; ++i)
        scanf("%d", &X[i]);

    int ps[N + 1];

    for (int i = 1; i <= N; ++i)
        ps[i] = ps[i - 1] + X[i];

    int resposta = 0;

    for (int a = 1; a <= N; ++a)
        for (int b = a; b <= N; ++b)
```

X

**Solução: Subtarefa 4 ( $A_i \leq 2$ )**

## Solução: Subtarefa 4 ( $A_i \leq 2$ )

★ Nesta subtarefa, todos candidatos tiraram ou nota 1 ou nota 2

## Solução: Subtarefa 4 ( $A_i \leq 2$ )

- ★ Nesta subtarefa, todos candidatos tiraram ou nota 1 ou nota 2
- ★ Só há duas alternativas para a nota de corte:  $C = 1$  e  $C = 2$

## Solução: Subtarefa 4 ( $A_i \leq 2$ )

- ★ Nesta subtarefa, todos candidatos tiraram ou nota 1 ou nota 2
- ★ Só há duas alternativas para a nota de corte:  $C = 1$  e  $C = 2$
- ★ Se a nota de corte for igual a 1, todos serão aprovados

## Solução: Subtarefa 4 ( $A_i \leq 2$ )

- ★ Nesta subtarefa, todos candidatos tiraram ou nota 1 ou nota 2
- ★ Só há duas alternativas para a nota de corte:  $C = 1$  e  $C = 2$
- ★ Se a nota de corte for igual a 1, todos serão aprovados
- ★ Se a nota de corte for igual a 2, apenas os candidatos que tiraram 2 serão aprovados



## Solução: Subtarefa 4 ( $A_i \leq 2$ )

- ★ Nesta subtarefa, todos candidatos tiraram ou nota 1 ou nota 2
- ★ Só há duas alternativas para a nota de corte:  $C = 1$  e  $C = 2$
- ★ Se a nota de corte for igual a 1, todos serão aprovados
- ★ Se a nota de corte for igual a 2, apenas os candidatos que tiraram 2 serão aprovados
- ★ Portanto, a resposta só será 2 quanto o número de candidatos que tiraram 2 for maior ou igual a  $K$ ; caso contrário, a resposta é igual a 1