



0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9

Utilize caneta azul ou preta e preencha completamente a quadrícula.
Exemplo: ■. Não use ☒.

Turma: (somente um número; consulte a pessoa responsável se não souber)

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------

← Marque as quadrículas ao lado para formar o seu número USP e escreva seu nome completo em letra legível na linha pontilhada abaixo. **Se seu número possui menos que 8 dígitos complete com zeros à esquerda.**

Nome: _____

Esta prova tem duração de 120 minutos. Não desmonte a prova.

Q1 [2 pontos] Simule o código abaixo e selecione as opções correspondentes a saída impressa do programa.

```
def f1(L, v, a, b, n):
    c = (a+b)//2
    print(a, c, b)
    if v == L[c]:
        return n-c
    elif a == c:
        return -1
    elif v > L[c]:
        return f1(L,v,a,c,n)
    elif v < L[c]:
        return f1(L,v,c,b,n)
    else: return -2
def f2(L, v, n):
    return f1(L, v, 0, 4, n)
print(f2([4,3,1,-5], -5, 3))
print(f1([11,5,4,1], 5, 2,5,5))
```

Rascunho

Selecione o primeiro número impresso:

<input type="checkbox"/> 0 1 2	<input type="checkbox"/> 0 2 2	<input type="checkbox"/> 0 4 8	<input type="checkbox"/> 1 4 8	<input type="checkbox"/> 1 4 6	<input type="checkbox"/> 1 2 4	<input type="checkbox"/> 0 2 4
--------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------

Selecione o segundo número impresso:

<input type="checkbox"/> 2 2 4	<input type="checkbox"/> 2 2 2	<input type="checkbox"/> 1 2 3	<input type="checkbox"/> 2 3 3	<input type="checkbox"/> 1 3 6	<input type="checkbox"/> 2 3 4	<input type="checkbox"/> 2 3 6
--------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------

Selecione o terceiro número impresso ou N/A (não aplicável) se não ocorre mais que 2 impressões:

<input type="checkbox"/> 3	<input type="checkbox"/> -3	<input type="checkbox"/> -2	<input type="checkbox"/> 1	<input type="checkbox"/> 0	<input type="checkbox"/> N/A	<input type="checkbox"/> -1	<input type="checkbox"/> 2
----------------------------	-----------------------------	-----------------------------	----------------------------	----------------------------	------------------------------	-----------------------------	----------------------------

Selecione o quarto número impresso ou N/A (não aplicável) se não ocorre mais que 3 impressões:

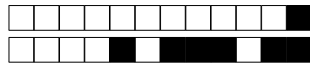
<input type="checkbox"/> 2 2 3	<input type="checkbox"/> 0 2 3	<input type="checkbox"/> 2 4 5	<input type="checkbox"/> 2 3 5	<input type="checkbox"/> 2 5 3	<input type="checkbox"/> N/A	<input type="checkbox"/> 2 5 5	<input type="checkbox"/> 2 3 3
--------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------	------------------------------	--------------------------------	--------------------------------

Selecione o quinto número impresso ou N/A (não aplicável) se não ocorre mais que 4 impressões:

<input type="checkbox"/> 2 5 6	<input type="checkbox"/> 2 3 5	<input type="checkbox"/> N/A	<input type="checkbox"/> 2 2 4	<input type="checkbox"/> 2 3 6	<input type="checkbox"/> 2 2 3	<input type="checkbox"/> 2 4 6	<input type="checkbox"/> 2 2 6
--------------------------------	--------------------------------	------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------	--------------------------------

Selecione o sexto número impresso ou N/A (não aplicável) se não ocorre mais que 5 impressões:

<input type="checkbox"/> 2	<input type="checkbox"/> 0	<input type="checkbox"/> -1	<input type="checkbox"/> N/A	<input type="checkbox"/> -3	<input type="checkbox"/> -2	<input type="checkbox"/> 3	<input type="checkbox"/> 1
----------------------------	----------------------------	-----------------------------	------------------------------	-----------------------------	-----------------------------	----------------------------	----------------------------



Q2 [2 pontos] Dada uma matriz quadrada A com dimensões $L \times C$ ($L \geq C$, linha x coluna). Considere que a matriz

	C=0	C=1	C=2	C=3
L=0	1	2	3	4
L=1	1	2	3	5
L=2	6	7	8	9
L=3	1	2	3	4

T1

T2

A é formada por dois triângulos, como indicado no exemplo da figura.

A função `minTriangulo(A)` deve criar uma nova matriz B , com dimensões $L \times C$, tendo como base as seguintes regras: - Os elementos do triângulo T2 da matriz B devem ser todos eles iguais ao menor elemento do triângulo T2 da matriz A ; - Os demais elementos da matriz B devem ser iguais a zero; Escolha os trechos de código a seguir que implementam corretamente a função `minTriangulo(A)`.

<pre># T1 - 0 nl = len(A); nc = len(A[0]); min = A[0][nl-1] # T2 - 0 nl = len(A); nc = len(A[0]); min = A[0][nc-1] # T3 - 0 for i in range(0,nl-1): # T4 - 0 for i in range(0,nl): # T5 - 1 for j in range(nc-1): # T6 - 1 for j in range(nl-i-1,nc): # T7 - 1 for j in range(nc): # T8 - 2 if (A[i][j]<min) : # T9 - 2 if (A[i][j]<A[i][j+1]):</pre>	<pre># T10 - 2 if (min<A[i][j]) # T11 - 2 if (A[i][j]<min) : min = A[i][j] # T12 - 2 if (A[i][j]<A[i][j+1]) : min = A[i][j] # T13 - 2 if (A[i][i+1]<min) : min = A[i][i+1] # T14 - 2 if (i < j) : linha = [] # T15 - 3 min = A[i][j] # T16 - 1 for j in range(nl)): # T17 - 1 for j in range(nl-1):</pre>	<pre># T18 - 0 B = [] # T19 - 0 for i in range(0,nl) : linha = [] # T20 - 0 linha = [] for i in range(0,nl) : linha.append(0) # T21 - 1 for j in range(0,nl-i-1) : linha.append(0) # T22 - 2 for j in range(nl-i-1,0): linha.append(min) # T23 - 1 for j in range(nl-i-1,nc): linha.append(min) # T24 - 1 B.append(linha) return B</pre>
---	---	--

Assinale a ÚNICA alternativa que contém os blocos corretos na ORDEM correta. Marcar mais de uma alternativa implica em ZERO.

- | | | |
|---|---|---|
| <input type="checkbox"/> 1,4,6,8,14,18,20,22,23,24 | <input type="checkbox"/> 1,4,6,8,15,18,19,21,23,24 | <input type="checkbox"/> 1,4,6,12,15,18,20,22,23,24 |
| <input type="checkbox"/> 2,4,6,8,14,18,20,22,23,24 | <input type="checkbox"/> 2,4,6,8,15,18,19,21,23,24 | <input type="checkbox"/> 1,3,6,13,15,18,20,21,23,24 |
| <input type="checkbox"/> 2,4,6,10,15,18,19,22,23,24 | <input type="checkbox"/> 2,4,6,11,15,18,19,22,23,24 | <input type="checkbox"/> 1,4,6,9,15,18,20,21,23,24 |
| <input type="checkbox"/> 1,4,6,10,15,18,20,22,23,24 | <input type="checkbox"/> 2,4,6,9,15,18,19,22,23,24 | <input type="checkbox"/> 2,3,6,8,14,18,19,21,23,24 |

Rascunho



Q3 [3 pontos] Preencha as lacunas no código acima (L1 até L9), de forma a obter um programa em Python que implemente a busca binária em uma matriz M de tamanho $m \times n$. Essa matriz possui duas propriedades: (i) os números em cada linha estão ordenados da esquerda para direita (ii) o primeiro número de cada linha é maior que o último número da linha anterior.

```
def busca(M, x):  
    col = len(M)  
    row = len(M[0])  
    start = 0  
    end = L1  
    L2  
    L3  
    L4  
    L5  
    return True  
    L6  
    L7  
    L8  
    L9  
    return False
```

Para cada lacuna, assinale aquele item que resulta em um programa correto. Cada lacuna erroneamente marcada poderá acarretar desconto.

L1:	<input type="checkbox"/> col * row + 1	<input type="checkbox"/> col * row - 1	<input type="checkbox"/> row	<input type="checkbox"/> col * row	<input type="checkbox"/> col * col	
L2:	<input type="checkbox"/> while start <= end:	<input type="checkbox"/> while start >= 0:	<input type="checkbox"/> while start >= end:	<input type="checkbox"/> while start >= row:	<input type="checkbox"/> while end >= 0:	
L3:	<input type="checkbox"/> mid = (row - start)//2	<input type="checkbox"/> mid = (col + row)//2	<input type="checkbox"/> mid = (col + row)	<input type="checkbox"/> mid = (start + end)	<input type="checkbox"/> mid = (start + end)//2	
L4:	<input type="checkbox"/> midValue = M[mid // end][mid % row]	<input type="checkbox"/> midValue = M[col][row]	<input type="checkbox"/> midValue = M[mid // col][mid % col]	<input type="checkbox"/> midValue = M[mid // col][mid % row]	<input type="checkbox"/> midValue = M[mid // row][mid % col]	<input type="checkbox"/> midValue = M[mid // row][mid % row]
L5:	<input type="checkbox"/> if end == start:	<input type="checkbox"/> if mid == x:	<input type="checkbox"/> if midValue == x:	<input type="checkbox"/> if M[mid][mid] == x:	<input type="checkbox"/> if M[midValue] == x:	<input type="checkbox"/> if midValue == start:
L6:	<input type="checkbox"/> if mid > x:	<input type="checkbox"/> if midValue > M[col][row]	<input type="checkbox"/> if mid > end:	<input type="checkbox"/> if midValue > M[mid][mid]:	<input type="checkbox"/> if mid > start:	<input type="checkbox"/> if midValue > x:
L7:	<input type="checkbox"/> end = row - 1	<input type="checkbox"/> end = row	<input type="checkbox"/> end = mid + 1	<input type="checkbox"/> end = mid - 1	<input type="checkbox"/> end = mid	
L8:	<input type="checkbox"/> else:	<input type="checkbox"/> elif midValue > x:	<input type="checkbox"/> elif mid > x:	<input type="checkbox"/> elif midValue >= x:	<input type="checkbox"/> elif mid == x:	
L9:	<input type="checkbox"/> start = end	<input type="checkbox"/> start = row	<input type="checkbox"/> start = mid	<input type="checkbox"/> start = mid + 1	<input type="checkbox"/> start = end + 1	



Q4 [3 pontos] Preencha as lacunas no código abaixo (L1 até L5), de forma a obter um programa que lê uma string e imprime todas as permutações possíveis entre os caracteres da string. Exemplo, se a string for ABC o programa deve imprimir ABC ACB BAC BCA CBA CAB

```
def permuta (a, l, r):  
    L1  
    print(a)  
    else:  
        L2  
        L3  
        L4  
        L5  
string = input('Digite a string')  
a = list(string)  
permuta(a, 0, len(a)-1)
```

Rascunho

Para cada um dos 5 itens a seguir, correspondendo as lacunas no código acima, assinale a única resposta que torna o programa acima correto.

L1:	<input type="checkbox"/> if len(l) == 0:	<input type="checkbox"/> if len(l) == len(r):	<input type="checkbox"/> if r == 0:	<input type="checkbox"/> if l == 0:
	<input type="checkbox"/> if l == r:	<input type="checkbox"/> if l < r:	<input type="checkbox"/> if l > r:	
L2:	<input type="checkbox"/> for i in range(0, l+1):	<input type="checkbox"/> for i in range(0, l):	<input type="checkbox"/> for i in range(l, r+1):	<input type="checkbox"/> for i in range(l, len(r)+1):
	<input type="checkbox"/> for i in range(l, r+1, -1):	<input type="checkbox"/> for i in range(0, r+1):	<input type="checkbox"/> for i in range(l+1, r):	
L3:	<input type="checkbox"/> a[i], a[r] = a[r], a[i]	<input type="checkbox"/> a[r], a[i] = a[i], a[r]	<input type="checkbox"/> a[l], a[i] = a[l+i], a[l]	<input type="checkbox"/> a[r], a[l] = a[l], a[r]
	<input type="checkbox"/> a[l], a[i] = a[i], a[l]	<input type="checkbox"/> a[l], a[r] = a[r], a[l]	<input type="checkbox"/> a[l+i], a[l] = a[l], a[l+i]	
L4:	<input type="checkbox"/> permuta(a, r+1, l)	<input type="checkbox"/> permuta(a, 0, r-1)	<input type="checkbox"/> permuta(a, 0, r+1)	<input type="checkbox"/> permuta(a, l+1, r+1)
	<input type="checkbox"/> permuta(a, l, r)	<input type="checkbox"/> permuta(a, l+1, r)	<input type="checkbox"/> permuta(a, l, r+1)	
L5:	<input type="checkbox"/> a[i], a[r] = a[r], a[i]	<input type="checkbox"/> a[r+i], a[l] = a[l], a[r+i]	<input type="checkbox"/> a[i], a[l] = a[l], a[i]	<input type="checkbox"/> a[r], a[l] = a[l], a[r]
	<input type="checkbox"/> a[l], a[i] = a[i], a[l]	<input type="checkbox"/> a[r], a[i] = a[i], a[r]	<input type="checkbox"/> a[l], a[r] = a[r], a[l]	