

PROVA 1
MAC0119 INTRODUÇÃO À PROGRAMAÇÃO DE COMPUTADORES (ICB)
2^o SEMESTRE DE 2025

Nome completo: _____

NUSP: _____

Instruções:

- (1) Esta prova é individual.
- (2) A prova consiste de **4** questões (contando a Questão 0 nesta página). Note que é possível tirar mais de 10 nesta prova :-)
- (3) Para ter nota integral em uma questão, a escrita de sua solução deve estar boa.
- (4) As respostas devem estar nos locais indicados.
- (5) Não é permitido o uso de aparelhos eletrônicos de qualquer natureza.
- (6) Não destaque as folhas deste caderno.
- (7) Não use folhas avulsas para rascunho. Não é necessário apagar seus rascunhos.
- (8) Não é permitido consultar colegas.
- (9) O único material que você pode consultar é seu *cheat sheet*.

Assinatura:

Sua assinatura acima atesta a autenticidade e originalidade de seu trabalho e que você compromete-se a seguir o código de ética da USP em todas as suas atividades, incluindo esta prova.

Boa sorte!

Q	0	1	2	3	Total
Nota					

Q0. [0.5 pontos] Leia o conteúdo desta página e preencha os itens requisitados. Assine acima, e atente ao significado de sua assinatura.

Data: 2025/10/22, 8:03pm

RASCUNHO

Q1. [3.5 pontos] Escreva abaixo seu nome completo (como consta no JupiterWeb) sem espaços e com todas as letras maiúsculas (por exemplo, se seu nome fosse Rosalind Franklin, sua resposta aqui seria ROSALINDFRANKLIN).

R: _____

Seja `s` o string que você escreveu acima. Diga o que é impresso quando são executados os comandos nos itens abaixo:

(i) `print(len(s))`

R: _____

(ii) `print(s[7])`

R: _____

(iii) `print(s[2 : 6])`

R: _____

(iv) `print(s[len(s) // 2])`

R: _____

(v) `print(s[len(s)//2 : 100])`

R: _____

(vi) `print(s[-1])`

R: _____

(vii) `print(s[2 : -2])`

R: _____

(viii) `print(s[-5:])`

R: _____

(ix) `print(s[0 : len(s) : 3])`

R: _____

(x) `print(s[7 : 0 : -1])`

R: _____

(xi) `print(s[:5] + s[7 : 12])`

R: _____

Q2. [4 pontos] Nesta questão, você precisa primeiro montar um string a partir de seu NUSP. Para tanto, sigas os passos abaixo.

(i) Seu NUSP: _____

(ii) Monte um número a partir de seu NUSP, concatenando seu NUSP com seu reverso. Por exemplo, se seu NUSP fosse 12345670, nesta etapa você montaria o número 1234567007654321. Faça este procedimento com seu NUSP. Qual é o resultado?

R: _____

(iii) Monte agora um string com as letras *a* e *b*, substituindo os dígitos pares de seu número do item (ii) por *a* e os ímpares por *b*. No caso do exemplo 1234567007654321, o resultado seria *babababaabababab*. Qual é o resultado deste processo com sua resposta para o item (ii) acima?

R: _____

Considere agora as seguintes funções:

```
# from the beginning up to 'ab' or 'ba'
```

```
def q2rest(s):
```

```
    i = 2
```

```
    while i < len(s):
```

```
        if s[i : i + 2] in ['ab', 'ba']:
```

```
            return s[:i]
```

```
        i += 2
```

```
    return ''
```

```
# starts with 'aa' or 'bb' and goes up to 'ab' or 'ba'
```

```
# frame: starting at 0 (2 letters at a time)
```

```
def q2orfs(s):
```

```
    orfs = []
```

```
    for i in range(0, len(s), 2):
```

```
        if s[i : i + 2] in ['aa', 'bb']:
```

```
            t = q2rest(s[i:])
```

```
            if t != '':
```

```
                orfs.append(t)
```

```
    return orfs
```

```
def q2longest(l):  
    t = ''  
    for s in l:  
        if len(s) > len(t):  
            t = s  
    return t
```

(iv) Suponha que `s` seja o string que você montou no item (iii) acima. O que será impresso pelo trecho de código abaixo?

```
print(q2rest(s))  
print(q2rest(s[1:]))
```

Resposta:

(v) Suponha que `s` seja o string que você montou no item (iii) acima. O que será impresso pelo trecho de código abaixo?

```
t = q2orfs(s)  
print(t)
```

Resposta:

- (vi) Suponha que `s` seja o string que você montou no item (iii) acima. O que será impresso pelo trecho de código abaixo?

```
tt = q2orfs(s[1:])  
print(tt)
```

Resposta:

- (vii) Continuando o código acima, o que será impresso pelo trecho de código abaixo?

```
# t e tt calculados acima  
print(q2longest(t + tt))
```

Resposta:

Q3. [4 pontos] Sejam L uma lista de strings (“padrões”) e s um string. Queremos saber quantas vezes os padrões em L ocorrem em s . Por exemplo, se

```
L = ['ab', 'ba']
```

```
s = 'abbabaabbaababba'
```

então há 10 ocorrências dos padrões em L em s (ab e ba ocorrem 5 vezes cada um em s). Nesta questão, você irá escrever uma função que, dados L e s , determina quantas vezes os padrões em L ocorrem em s . Você deverá fazer isso seguindo as instruções abaixo.

- (i) Escreva inicialmente uma função chamada `count_pattern()` que recebe um padrão p um string s e que devolve o número de vezes que p ocorre em s . Por exemplo, a execução `count_pattern('ab', 'abbabaabbaababba')` deve devolver 5. Escreva aqui sua função `count_pattern(p, s)`.

Resposta:

- (ii) Escreva agora uma função chamada `count()` que recebe uma lista de padrões `L` e um string `s` e que devolve o número de vezes que os padrões em `L` ocorrem em `s`.

Por exemplo, a execução `count(['ab', 'ba'], 'abbabaabbaababba')` deve devolver 10. Escreva aqui sua função `count(L, s)`.

Observação. Use, obrigatoriamente, a função `count_pattern()` de (i) acima, mesmo que você não a tenha escrito.

Resposta:

* * *

Rascunho:

A large, empty rectangular box with a thin black border, intended for a draft or sketch. It occupies the majority of the page area below the 'Rascunho:' label.

Rascunho:

A large, empty rectangular box with a thin black border, occupying the majority of the page below the 'Rascunho:' label. It is intended for a draft or sketch.