

Aula 26: Vetor de contagem por letra

Lógica de Programação – Superior · Bloco 3

José Dihego · IFBA

O que você precisa saber hoje

Objetivo: Indexar um vetor de 26 posições pela letra ($c - 'A'$).

- Índice da letra: $c - 'A'$ (maiúscula) ou $c - 'a'$ (minúscula), de 0 a 25
- Volta: $'A' + k$ é a letra de índice k
- Contagem: $\text{cont}[k]++$; primeira posição: guardar só se ainda for -1
- Inicialize com um laço quando o valor inicial não for 0

```
int primeira[26];
for (int k = 0; k < 26; k++) primeira[k] = -1;
for (int i = 0; s[i] != '\0'; i++) {
    int k = -1;
    if (s[i] >= 'A' && s[i] <= 'Z') k = s[i] - 'A';
    if (s[i] >= 'a' && s[i] <= 'z') k = s[i] - 'a';
    if (k != -1 && primeira[k] == -1) primeira[k] = i;
}
```

Erros que custam pontos

Errado	Certo	Por quê
<code>cont[s[i]]++;</code>	<code>cont[s[i] - 'A']++;</code>	índice 97 não existe num vetor de 26
<code>int primeira[26] = {-1};</code>	laço preenchendo -1	só a posição 0 recebe -1
<code>cont[s[i] - 'A']++; /* com espaço */</code>	<code>/* teste a faixa antes */</code>	' ' - 'A' é negativo

```
char s[] = "Oba, boa"; int primeira[26];
for (int k = 0; k < 26; k++) primeira[k] = -1;
for (int i = 0; s[i] != '\0'; i++) {
    int k = -1;
    if (s[i] >= 'A' && s[i] <= 'Z') k = s[i] - 'A';
    if (s[i] >= 'a' && s[i] <= 'z') k = s[i] - 'a';
    if (k == -1) continue;
    if (primeira[k] == -1) primeira[k] = i;
}
```

i	s[i]	k	primeira[k]
0	O	14	0
1	b	1	1
2	a	0	2
5	b	1	1
6	o	14	0
7	a	0	2

Avaliação 3 – 2024.2, Questão II

avaliacao_3_ano_2024_2.pdf

Posição da primeira ocorrência de cada letra

Estratégia:

1. Ler a linha com fgets
2. Para cada letra, guardar a posição só na primeira vez
3. Imprimir em ordem alfabética as letras que apareceram

Enunciado completo (com figuras e exemplos) no PDF da prova.

Solução (1/2)

```
1  #include <stdio.h>                // printf e fgets
2
3  int main(void) {                  // início do programa
4      char s[1001];                 // a string lida
5      printf("Texto: ");            // pede o texto
6      fgets(s, sizeof s, stdin);    // lê a linha inteira
7      int primeira[26];             // primeira[k]: posição da 1ª ocorrência da letra 'A'
          + k
8      for (int k = 0; k < 26; k++) primeira[k] = -1; // -1 = ainda não apareceu
9      for (int i = 0; s[i] != '\0'; i++) { // percorre o texto
10         int k = -1;                // índice da letra -(1: não é letra)
```

Solução (2/2)

```
11 -
12     if (s[i] >= 'A' && s[i] <= 'Z') k = s[i] - 'A'; // maiúscula: '0' - 'A' = 14
13     if (s[i] >= 'a' && s[i] <= 'z') k = s[i] - 'a'; // minúscula: 'o' - 'a' = 14
14     if (k != -1) {                                // só letras (sem acento)
15         if (primeira[k] == -1) primeira[k] = i; // guarda só a primeira vez
16     }
17 }
18 for (int k = 0; k < 26; k++)                        // de A a Z, em ordem
19     if (primeira[k] != -1) printf("%c(%d) ", 'A' + k, primeira[k]); // só as letras que
    apareceram
20 printf("\n");                                       // termina a linha
21 return 0;                                           // fim sem erro
22 }
```

Avaliação 3 – 2026.1, Questão II

avaliacao_3_ano_2026_1.pdf

Assinatura de frequência (2N3A1B aceita BANANA)

Dicas:

- Conte as letras da palavra num vetor de 26
- A fórmula vem em pares dígito + letra: `exigido[letra - 'A'] = dígito - '0'`
- Todas as letras precisam bater

Para conferir: entrada 2N3A1B BANANA → a saída deve conter ACEITO

A solução comentada está na página do curso: <https://josedihego.net/logica/#aula-26>