

Aula 33: Busca em grade de caracteres

Lógica de Programação – Superior · Bloco 3

José Dihego · IFBA

O que você precisa saber hoje

Objetivo: Procurar padrões em linhas e colunas de uma grade.

- Para cada posição inicial, compare a palavra letra a letra
- Horizontal: $g[i][j + k]$; vertical: $g[i + k][j]$
- Cheque antes se a palavra cabe: $j + t \leq n$ e $i + t \leq m$
- Por colunas: percorra k nas linhas fixando a coluna

```
int h = (j + t <= n), v = (i + t <= m);  
for (int k = 0; k < t; k++) {  
    if (h && g[i][j + k] != p[k]) h = 0;  
    if (v && g[i + k][j] != p[k]) v = 0;  
}  
if (h || v) achou = 1;
```

Erros que custam pontos

Errado	Certo	Por quê
<code>g[i][j + k] sem testar j + t <= n</code>	<code>h = (j + t <= n)</code>	leitura fora da linha
<code>achou = (g[i][j] == p[0]);</code>	<code>/* compare todas as letras */</code>	só a primeira letra não basta

```
const char *g[3] = {"GATO", "AXIS", "SOLO"};
const char *p = "GAS"; int t = 3;
for (int i = 0; i < 3; i++)
    for (int j = 0; j < 4; j++) {
        if (g[i][j] != p[0]) continue;
        int h = j + t <= 4, v = i + t <= 3;
        for (int k = 0; k < t; k++) { if (h && g[
i][j + k] != p[k]) h = 0; if (v && g[i + k][
j] != p[k]) v = 0; }
    }
```

(i, j)	horizontal	vertical
(0, 0)	não	sim

Avaliação 3 – 2023.1, Questão I

avaliacao_3_ano_2023_1.pdf

Caça-palavras (horizontal e vertical) – pontuação

Estratégia:

1. Ler a grade e a palavra
2. Tentar cada posição em horizontal e vertical
3. Pontuação: tamanho da palavra ou -5

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

Solução (1/2)

```
1  #include <stdio.h>           // printf e scanf
2  #define MAX 100             // maior grade aceita
3
4  int main(void) {             // início do programa
5      int m, n;                // linhas e colunas da grade
6      char g[MAX][MAX + 1], palavra[MAX + 1]; // grade e palavra procurada
7      scanf("%d %d", &m, &n);   // lê as dimensões
8      for (int i = 0; i < m; i++) scanf("%s", g[i]); // lê as linhas da grade
9      scanf("%s", palavra);     // lê a palavra
10     int t = 0, achou = 0;      // tamanho da palavra e resultado
11     while (palavra[t] != '\0') t++; // conta as letras até o '\0'
12     for (int i = 0; i < m && !achou; i++) { // cada posição inicial
13         for (int j = 0; j < n && !achou; j++) { // (para assim que achar)
```

Solução (2/2)

```
14         int h = j + t <= n, v = i + t <= m;  // cabe na horizontal? na vertical?
15         for (int k = 0; k < t; k++) {        // compara letra a letra
16             if (h && g[i][j + k] != palavra[k]) h = 0;  // falhou na horizontal (anda pela
coluna)
17             if (v && g[i + k][j] != palavra[k]) v = 0;  // falhou na vertical (anda pela
linha)
18         }
19         if (h || v) achou = 1;                // encontrou em alguma direção
20     }
21 }
22 printf("%d pontos\n", achou ? t : -5);  // tamanho da palavra ou -5
23 return 0;                                // fim sem erro
24 }
```

Avaliação 3 – 2023.1, Questão II

avaliacao_3_ano_2023_1.pdf

Consenso de cinco fitas de DNA

Dicas:

- As 5 fitas formam uma matriz 5×8
- Para cada coluna, conte A, C, G e T
- A mais frequente vence; empate vira ?

Para conferir: entrada CGGATTTT CGGAAATT GCGTTAAT GCGTTTTT ACGGACAT → a saída deve conter ?CG?T?TT

A solução comentada está na página do curso: <https://josedihego.net/logica/#aula-33>