

O trabalho pede para fazer um jogo do tipo caça palavras, em que é gerada uma matriz 10x10 aleatória (a partir de uma semente). O usuário deverá então inserir palavras no programa que procurará se essa palavra está na matriz, analisando ela na vertical, horizontal e diagonal.

Para resolver o problema foram usados os recursos ensinados em aula assim como encontrados na internet.

Primeiramente para simplificar o problema ele foi dividido em partes e subpartes:

- 1) Gerar a Caça-palavras:
  - a) Percorrer a matriz para: dar valores e imprimir a própria.
  - b) Gerar números randômicos a partir de uma semente.
- 2) Percorrer a matriz
  - a) Como percorrer a matriz na vertical
  - b) Como percorrer a matriz na diagonal
- 3) Achar as palavras na matriz
  - a) Conferir se a palavra não está dentro da matriz

Resolução:

1.a) Para percorrer a matriz foi usado dois “for loops” aninhados um que percorria as linhas e outro dentro dele para percorrer todas as colunas daquela linha, assim foi possível dar valores e imprimi-la.

1.b) Para gerar os números randômicos a partir de uma semente foi usada a função `srand` e `rand`, que estão na biblioteca `stdlib.h`. Os números gerados foram então convertidos para ir até 26 algarismos e serem, dentro da lista de `int` para `char`, apenas letras do alfabeto.

2.a) Para percorrer a matriz na vertical foi apenas inverter a ordem em que aconteciam os loops aninhados da horizontal.

2.b) A diagonal foi percorrida usando 2 loops aninhados, a maneira que isso foi feito foi: Primeiro percorrendo a diagonal principal. A partir disso para percorrer o resto da matriz foi simples, apenas mover a coluna 1 casa para direita e rodando o mesmo loop e assim por diante até acabar as colunas, e depois o mesmo foi feito denovo só que agora movendo a linha 1 casa para baixo ao invés da coluna.

3.a) Para achar as palavras na matriz foi usada funções que comparam se a palavra digitada está dentro da matriz com o uso de um `if` dentro de um loop, essa função é usada dentro do loop que percorre a matriz em cada direção.

3.b) Para conferir se a palavra não estava dentro da matriz foi usada uma variável de controle e um `IF` para conferir essa variável

A solução do trabalho foi obtida a partir de pesquisas, dúvidas tiradas com a professora, e conhecimentos já adquiridos na aula, os resultados finais foram os mesmos pedidos pelo trabalho. O código depois de pronto foi alterado para otimização, reuso e

facilitação de leitura, sendo algumas partes separadas em funções e valores como tamanho da matriz trocados por constantes. A indentação e comentários foram usados corretamente durante a realização do trabalho para melhor comunicação dos participantes.

**fontes:**

função rand e srand = [https://mazer.dev/tutorial-para-geracao-de-numeros-aleatorios-randomicos-em-](https://mazer.dev/tutorial-para-geracao-de-numeros-aleatorios-randomicos-em-c/#:~:text=Para%20gerar%20n%C3%BAmeros%20pseudo%2Daleat%C3%B3rios,RAND_MAX.)

[c/#:~:text=Para%20gerar%20n%C3%BAmeros%20pseudo%2Daleat%C3%B3rios,RAND\\_MAX.](https://mazer.dev/tutorial-para-geracao-de-numeros-aleatorios-randomicos-em-c/#:~:text=Para%20gerar%20n%C3%BAmeros%20pseudo%2Daleat%C3%B3rios,RAND_MAX.)

lista de inteiros para char = slide dado pela professora: Aula 2-3