

Exercício prático.

Escreva um programa que manipula matrizes com alocação “just in time”.

Deve existir uma estrutura com a seguinte configuração:

```
struct Mat {
    int linhas;
    int colunas;
    float **dados;
};
typedef struct Mat Matriz;
```

E as seguintes funções:

```
// Cria uma estrutura Matriz e aloca apenas o vetor de ponteiros para as linhas.
Matriz criaMatriz(int linhas, int colunas);

// Verifica se a linha passada já foi alocada e, se não foi, aloca antes de
atribuir o valor à linha e coluna especificados.
void setMatriz(int linha, int coluna, float valor, Matriz matriz);

// Verifica se a linha já foi alocada e, caso verdadeiro, retorna o dado na
posição linha e coluna da matriz, caso falso, retorna 0.
float getMatriz(int linha, int coluna, Matriz matriz);

// Imprime a matriz na tela.
void imprimeMatriz(Matriz matriz);

// Exemplo de uso.
int main(void) {
    puts("Matriz dinâmica"); /* prints !!!Hello World!!! */
    Matriz matriz;
    matriz = criaMatriz(3,3);
    setMatriz(0,0,1,matriz);
    printf("%f\n", getMatriz(0,0, matriz));
    printf("%f\n", getMatriz(1,0, matriz));
    imprimeMatriz(matriz);
    return EXIT_SUCCESS;
}
```