# Relatório de Algoritmo e Estrutura de Dados 1

fevereiro 2023

Grupo

|  |
| --- |
| **Grupo [nome completo dos integrantes]:** |
| Richard Silva e Souza |
|  |
|  |
|  |
|  |

Projeto GitHub

|  |
| --- |
| **Link para o Projeto GitHub utilizado para o Relatório:** |
| https://github.com/kitsunekamisama/TADSllist/blob/3095efef43ef6fc4abe336bf2d30c4fb84e49ded/main.c |

Exemplos de Código no Projeto GitHub (com nome do arquivo e número das linhas) que evidenciam os CONCEITOS presentes nos títulos de cada seção a seguir:

|  |
| --- |
| **Algoritmo de pesquisa (ou Busca):** |
| main.c Na linha 101, é utilizado o algoritimo de busca linear para encontar o elemento de índice ‘pos’ na lista |

|  |
| --- |
| **Algoritmo de ordenação:** |
| Esse código não possui algoritmo de ordenação. |

|  |
| --- |
| **Tipo Abstrato de Dados (TAD):** |
| main.c A partir da linha 6, é definido o TAD ‘Tno’, que representa um nó de lista encadeada.  A partir da linha 14, é definido o TAD ‘TLista’, que representa uma lista encadeada também. |

|  |
| --- |
| **Estrutura de Dados:** |
| main.c A partir da linha 6, é definido o TAD ‘Tno’, que representa um nó de lista encadeada.  A partir da linha 14, é definido o TAD ‘TLista’, que representa uma lista encadeada também.  Nas linhas 32 e 33, é criada uma variável do tipo ‘Tlista’, que representa uma lista encadeada. |

|  |
| --- |
| **Alocação Dinâmica da Memória:** |
| main.c Na linha 24, é feita a alocação dinâmica de um novo nó da lista.  Na linha 101, é feita a alocação dinâmica de um array de inteiros para armazenar a entrada do usuário.  Na linha 112, é feita a alocação dinâmica de um novo nó da lista. |

|  |
| --- |
| **Se houver alguma observação a fazer, o Grupo pode utilizar este espaço:** |
| Não consegui encontrar um código que continha dentro da sua estrutura todos os conceitos, faltando o algoritmo de ordenação. |