

• Definindo a estrutura do nó:

```
typedef struct Node {  
    int data;
```

• Dados do nó:

```
    struct Node * next; } Node;
```

• Função para criar um novo nó:

```
Node* createNode (int data) {
```

```
    Node* newNode =
```

```
    (Node*) malloc (sizeof (Node));
```

```
    if (newNode == NULL) {
```

```
        printf ("Erro na alocação de memória.");
```

```
        exit (EXIT_FAILURE); }
```

```
    newNode->data = data;
```

```
    newNode->next = NULL;
```

```
    return newNode; }
```

• Função de busca sequencial:

```
Node* current = head;
```

```
while (current != NULL) {
```

```
    if (current->data == target) {
```

```
        return current; }
```

```
    current = current->next; }
```

```
return NULL; }
```

• int main() { // Inicializando a lista vazia //

```
    Node* head = NULL;
```

```
    // Inserindo elementos no início da lista
```



D	S	T	Q	Q	S	S
D	L	M	M	J	V	S

```
head = insertAtBeginning(head, 3);
head = insertAtBeginning(head, 7);
head = insertAtBeginning(head, 1);
```

// Imprimindo a lista

```
Printf("Lista:");
PrintfList(head);
```

// Exemplo de busca sequencial

```
int target = 7;
```

```
Node* result = search(head, target);
```

```
if (result != NULL) {
```

```
    Printf("Elemento %d
```

```
    encontrado na lista.\n", target);
```

```
} else {
```

```
    Printf("Elemento %d não
```

```
    encontrado na lista.\n", target); }
```

// Liberando a memória alocada

```
FreeList(head);
```

```
return 0; }
```



O código apresenta uma implementação de lista encadeada em 'C' onde a estrutura 'node' representa as elementos da lista. A função 'createNode' aloca dinamicamente um novo nó e inicializa seus campos. A 'insertAtBeginning' adiciona um novo nó no início da lista, e 'PrintList' imprime os elementos. A 'freeList' libera a memória alocada para os nós.

No 'main()', a lista é inicializada, elementos são inseridos, a lista é impressa e, ao final, a memória é liberada. A função 'search' realiza busca sequencial na lista, retornando o valor desejado com os dados de cada nó. No exemplo do 'main()', ele busca o elemento 7 e indica se foi encontrado ou não. Em resumo, o código demonstra a manipulação de uma lista encadeada, incluindo inserção, impressão, busca e liberação de memória.