



Data Structures

HW2

Sina Rostami
9822143

Fall 2020

Problem 1:

we want to :

1. copy the whole linked-list's nodes in an array.
2. delete the ex-linked-list.
3. create a new linked-list and copy the nodes of the array to it from the end.

Pseudo-Code

```
let linked_list be the linked-list we want to reverse.
let size be the size of the linked_list.
let arr be an array of data with length := size.
let temp_iterator points to head of linked_list.
for i in range(0, size - 1) do
    | arr[i] ← temp_iterator.data
    | temp_iterator ← temp_iterator.next
end
let new_linket_list be a linked-list.
for i in range(size - 1, 0) do
    | Push(new_linket_list, arr[i])
end
Function Push(list, data):
    | if list.head equals with NULL then
        | list.head ← new Node(data)
        | return
    | end
    | let temp_iterator points to head of list
    | while temp_iterator.next not equals with NULL do
        | temp_iterator ← temp_iterator.next
    | end
    | temp_iterator.next ← new Node(data)
return
.
```

Now the *new_linket_list* is the linked-list we want.