

Lab 5.3.3 Singly linked list: part 3

Objectives

Familiarize the student with:

- implementing data structures in C++;
- providing derived data about the implemented data structure.

Scenario

Let's continue working on our list.

One piece of information that might be useful is the size of the list.

Add the method `size()` to your list implementation.

As the name suggests, it should return the number of stored elements.

```
#include <iostream>

using namespace std;

class List
{
public:
    List();
    ~List();
    void push_front(int value);
    bool pop_front(int "value");
    int size();
private:
    Node* head;
};

// ...

int main()
{
    List list;
    for (int i = 0; i <= 5; i++)
    {
        list.push_front(i);
        cout << "pushed " << i << ", size: " << list.size() << endl;
    }
    cout << endl;
    for (int i = 0; i <= 3; i++)
    {
        int value;
        list.pop_front(i);
        cout << "popped " << i << ", size: " << list.size() << endl;
    }
    return 0;
}
```

Example input

Example output

```
pushed 0 size: 1  
pushed 1 size: 2  
pushed 2 size: 3  
pushed 3 size: 4  
pushed 4 size: 5  
pushed 5 size: 6
```

```
popped 5 size: 5  
popped 4 size: 4  
popped 3 size: 3
```