Send an email with the solutions as attachments (.c, .cpp or if you have .h and .hpp files) to the following email address: tofihomework+2024info3en@gmail.com

You shouldn't attach the compiled programs. If you feel like you're sending a lot of files (5+) you can put them in a zip, but you don't need to.

LIFO

Implement a LIFO (last-in-first-out) data structure:

- Each "http://wiki.math.bme.hulink"http://wiki.math.bme.hu should store an integer.
- It should have a **push** function, that adds a new element at the beginning.
- And a **pop** function, that removes the first element and returns its stored value.
- Write a **main** function to test it!

Hints at the end of the page (if you have the time/perseverance, first try to solve it without the help, or just think about how you would solve it).

The data structure is basically the same as the linked list we used on the lecture. The only difference is that this time we're adding elements to the beginning of the list instead of the end. But still we can use a similar structure with a **next** pointer.

The **push** will be easier to write compared to the append for linked lists. We don't need to navigate to end of the list this time. We only need to set the **start** pointer to the new element and set its **next** pointer to the previous first element. The head of the **push** function could look something like this:

```
void push(struct lifo_e **start, int n);
```

The **pop** won't be too complex either. The **start** pointer stores the value to be returned so we just need to store that and its **next** pointer. Then we can delete it. Don't forget to set the **start** pointer to the next element before returning the stored value. The head of the **pop** function could look something like this:

```
int pop(struct lifo_e **start);
```