Exercises — Stack me baby

version #

Assistants C/Unix 2022 <assistants@tickets.assistants.epita.fr>

Copyright

This document is for internal use at EPITA (website) only.

Contents

*https://intra.assistants.epita.fr

1 Stack me baby

Files to submit:

- stack/stack.c

Provided files:

- stack/stack.h

Authorized functions: You are only allowed to use the following functions:

- malloc(3)
- calloc(3)
- free(3)

Authorized headers: You are only allowed to use the functions defined in the following headers:

- errno.h
- assert.h
- err.h
- stddef.h

1.1 Goal

You have to implement a stack (a LIFO data structure) of integers.

The structure used for this exercise is the following:

```
struct stack
{
    int data;
    struct stack *next;
};
```

1.1.1 Insertion

```
struct stack *stack_push(struct stack *s, int e);
```

This function adds the `e` element on top of the stack `s` and returns the top of the stack after the insertion. If any error occurs, you have to return `s` as it was when your function was called.

Be careful!

Calling `stack_push(NULL, 42)` is not an error, it should return a stack with a single element, 42.

1.1.2 Remove

```
struct stack *stack_pop(struct stack *s);
```

This function removes the top of the stack `s` and returns the new top. If `s` is `NULL`, returns `NULL`.

1.1.3 Get Value

```
int stack_peek(struct stack *s);
```

This function returns the data stored at the top of the stack. You do not have to handle the case where `s` is `NULL`.

It is my job to make sure you do yours.