**Secure and Dependable Systems**
Constructor University
Dr. Jürgen Schönwälder

Module: CO-566
Date: 2024-02-26
**Due: 2024-03-04**

# Problem Sheet #3

## Problem 3.1: *stack frames* (10 points)

The following C program calculates the length of the longest palindromic subsequence or a string.

```c
#include <string.h>                                         /* lps.c */
#include "lps.h"

static int lps_range(char *s, char *e)
{
    int n1, n2;

    if (s > e) {
       return 0;
    }

    if (s == e) {
       return 1;
    }

    if (*s == *e) {
        return 2 + lps_range(s+1, e-1);
    }

    n1 = lps_range(s+1, e);
    n2 = lps_range(s, e-1);
    return n1 > n2 ? n1 : n2;
}

int lps(char *s)
{
    size_t len = strlen(s);

    return lps_range(s, s+len-1);
}
```

The main function reads strings from the standard input and writes the strings together with the length of the longest palindromic subsequence to the standard output.

```c
#define _POSIX_C_SOURCE 200809L
#include <stdio.h>                                          /* main.c */
#include <stdlib.h>
#include "lps.h"

int main(void)
{
    char *line = NULL;
    size_t linecap = 0;
    ssize_t linelen;

    while ((linelen = getline(&line, &linecap, stdin)) > 0) {
        if (line[linelen-1] == '\n') {
            line[linelen-1] = 0;
        }
        printf("%s:\t%d\n", line, lps(line));
    }
    if (line) {
        free(line);
    }
21      return EXIT_SUCCESS;
22  }
```

Assume the programs reads the string `better` and the execution stops when the recursion producing the right solution hits the innermost base case. How does the call stack look like at this point in time? Complete the following figure:

```
heap:            "better"
                  ^                 :                 :   :
                  |                 +-----------------+   |
...        <------------------------| return address  |   |
                  |                 | frame pointer   |---'
                  '-----------------| char *line      |<--.
                                    | size_t linecap  |   |
                                    | ssize_t linelen |   |
                                    +-----------------+   |
 main.c:15 <------------------------| return address  |   |
```

The string `better` is stored on the heap. The stack frame of the main function is shown and the beginning of the next stack frame. The return address of this incompletely shown stack frame points to line 15 of the file main.c (we omit the column offset). Visualize all pointers when you add additional stack frames.