Code decomposition grading

1) The main project constraint is a maximum of 40 lines/function. The grade is purely based on the quantitative criteria of function size as detailed below whereas the comments analyze additional choices such as efficiency of parameter transmission and can make suggestions.
(a) If there is only "main() and the provided error functions" => grade = 0 pts
(b) Otherwise, if main() is too large, remove 1 pt.
(c) Otherwise each too large function removes 0.5 pt as long as the grade is positive.

CODE STYLE CRITERIA LIST BASED ON SOME **[CONVENTIONS]**

[L1] missing indentation for control statement (`if, for, while`...) or the indentation style is not constant over the whole code written by the student. PLEASE first check [L11] that indicate we accept 3 more indentation styles of *single controlled instruction*

[L11] missing indentation for the body of any function.

[L14] double indentation in a block of code (too much indentation)

```
for(i=0 ; i< MAX ; i++)
   {
        printf("this is doubly indented\n");
   }
```

[L2] there are at least TWO lines beyond the maximum size of 87 (= wrapping)

[L22] a long instruction or function declaration/definition/call organized on more than one line must *align with element of the previous line* that makes it readable, for example with the start of the parameter list for function calls, or with the start of the evaluated expression in an "if".

```
if(nb_robot > 0 && nb_obstacle > 0)
    deplace_robot( tab_robot, nb_robot,
                   tab_obstacle, nb_obstacle);
```

[R2] there is a global variable that is not a constexpr or a symbol (define, enum)

[P5] Magic numbers. Here are the contexts that were discussed on the project forum and are NOT penalized:

- image array index starting at 0 or 1.
- The "+1" or "-1" in expressions to access the neighbor pixels
- Initial value of 0 for counters

The following contexts are penalized:
1) **missing symbols** (with define or enum or constexpr). Focus on the magic numbers for the size of fixed arrays such as for the color components or explicit color index,…

2) **poor choice of symbol name** such as ZERO, ONE, TWO… for 0, 1, 2… showing that such a symbol cannot be a *parameter*.

For full details: refer to the course coding convention is visible on moodle