Modularité (exemple)

stack.h

```
#ifndef STACK_H
#define STACK_H

struct stack;
typedef struct stack *stack;

/* create an empty stack */
extern stack stack_create(void);

/* destroy a (not necessarily empy) stack */
extern void stack_destroy(stack s);

/* return a true value iff the stack is empty */
extern int stack_empty(stack s);

/* push an element on top of the stack */
extern void stack_push(stack s, void *element);

/* pop the top element of the stack and return it */
extern void *stack_pop(stack s);

#endif
```

éviter plusieurs inclusions du fichier dans un programme

il y a une struct stack, mais on ne connait pas les détails

le type principal manipulé sera un pointeur

expliquer le protocole avec des commentaires

pour indiquer zéro paramètres

expliciter que c'est l'interface

utilisation de la forme impérative du verbe pour avoir moins de texte

mettre le nom des paramètres

utilisation du pointeur générique pour les éléments

éviter plusieurs inclusions du fichier dans un programme