

Série d'exercices #8 $\frac{1}{2}$

IFT-2035

November 17, 2024

Programmation fonctionnelle en C

Soit la fonction C suivante:

```
void main (void)
{ /* Élimine les caractères répétés et
  * stoppe après la première ligne vide. */
  int c;
  int last = EOF;
  while ((c = getchar ()) != EOF) {
    if (c == last) {
      if (c == '\n') {
        break;
      } else {
        continue;
      }
    }
    putchar (last = c);
  }
}
```

D'abord, réécrire le code dans un style de programmation structurée stricte, c'est à dire sans utiliser de `continue`, `break`, ou `goto`.

Ensuite, réécrire le code à nouveau mais cette fois dans un style fonctionnel, c'est à dire sans opération d'affectation (sauf bien sûr dans les initialisations, e.g. `int last = EOF`). Il faudra introduire une fonction récursive auxiliaire et éliminer `while`, `break`, et `continue`.

Essayer de faire ces réécritures en les subdivisant en plusieurs petits pas simples, où on peut facilement voir que chaque pas préserve la sémantique du code précédent, sans avoir besoin de comprendre le code dans son ensemble.