

Exercices INF8541: série #7

1.

Les opérations suivantes définissent brièvement l'API des I-structures (pour plus de détails, voir l'énoncé du devoir #4):

```
typedef struct {
    bool vide;           # Indique si la cellule est vide ou non.
    T contenu;           # Valeur conservée par la cellule (si pas vide).
} I_struct_item;

typedef I_struct_item* I_STRUCTURE;

void I_ALLOC( int nb, I_STRUCTURE* i_s );
void I_READ( I_STRUCTURE i_s, int index, T* val_lue );
void I_WRITE( I_STRUCTURE i_s, int index, T val, bool* ok );
void I_FREE( I_STRUCTURE* i_s );
```

En utilisant une I-structure, écrivez (dans la notation d'Andrews) une procédure `proc` qui va définir un processus `producteur` et un processus `consommateur` tel que le `producteur` va générer une série aléatoire de nombres entiers, lesquels seront transmis au `consommateur` qui les imprimera.

L'interface des différentes procédures sont les suivantes:

```
void proc( int nb_elements );
/* Cree les processus producteur et consommateur en leur
   transmettant les informations appropriées, dont nb_elements,
   le nombre d'elements a generer et a copier. */

void producteur( int nb, ... );
/* Genere les elements et les transmet, par l'intermediaire d'une
   I-structure appropriée. */

void consommateur( int nb, ... );
/* Recoit les elements generes puis les imprime sur stdout. */
```