



## Journée 1 - TP 2

*BSP UBoot et Linux embarqué, construire son propre système*

*Date MAJ*

*22 août 2023*

*Chiheb Ameer ABID*

*Version*

*1.0*

### Exercice 1.

- Créer un module qui sera paramétré comme suit :
  - Lors du chargement :
    - Il va créer trois threads successifs.
    - Les threads font appel à une fonction définie qui affichent respectivement :
      - Une chaîne de caractères saisie en entrée.
      - La différence entre deux nombres saisis en entrée
      - La somme de deux nombres saisis en entrée.
  - Lors du déchargement :
    - Il va arrêter tous les threads créés et afficher un message de clôture.

### Exercice 2.

Reprendre l'exercice de TP1 pour faire clignoter les diodes LED. L'appuie sur un bouton poussoir active ou désactive le clignotement de la diode LED associée.

### Exercice 3.

On cherche à implanter un jeu de « nombre mystère fuyant » par une module noyau. Il s'agit d'une variante du jeu du nombre mystère où l'ordinateur choisit un nombre entier aléatoirement et ne répond aux propositions d'un joueur que par Trop petit !, Trop grand ! ou Gagné !.

Dans cette variante, toutes les  $t$  secondes l'ordinateur change le nombre mystère en lui ajoutant ou en lui retirant un nombre  $x$ . Le joueur en est informé par un message (par exemple : Le nombre mystère a été augmenté de 13 !). Les nombres  $t$  et  $x$  sont définis aléatoirement et changent à chaque fois qu'on les utilise (par exemple après 5 secondes de jeu l'ordinateur ajoute 32 au nombre mystère, puis au bout de 11 secondes, il lui retire 13, etc.). Le joueur n'aura de plus qu'un temps limité pour trouver le nombre fuyant. Réalisez l'application implantant le jeu du nombre fuyant à l'aide de trois threads.

Le thread initial réalisera le jeu du nombre mystère classique : le choix d'un nombre aléatoire. Un premier thread annexe se chargera des modifications du nombre mystère dans le temps. Un second thread annexe se chargera du respect du temps limite.

Pour l'implantation, le nombre mystère sera choisi entre 0 et 200,  $t$  entre 5 et 10,  $x$  entre 0 et 50 sera ajouté ou retiré (choix aléatoire) avec la contrainte de préserver le nombre mystère entre 0 et 200, enfin, le temps limite sera de 40 secondes. Le module informera l'utilisateur du temps du jeu à la fin d'une partie.

L'interaction depuis l'espace utilisateur avec le jeu se fait de la manière suivante :

- Le démarrage du jeu, s'effectue en écrivant 'run' dans le fichier associé au module
- On peut forcer l'arrêt du jeu en écrivant 'stop' dans le fichier associé au module
- Une proposition de joueur s'effectue en écrivant dans le fichier du module