

TD architectures pseudo-parallèles DEUST 2

03/12/2020

H. TSOUNGUI

Gestion des entrées et sorties dans un parking public

La ville de Perpignan souhaite proposer aux usagers d'un parking public, **un outil visuel permettant de disposer, en temps réel, d'informations sur** les occupations de son parking. Le Responsable de la commune vous demande de développer un programme si possible temps réel, permettant de visualiser la situation à chaque instant.

Quelques données à connaître :

-Les arrivées par les entrées E1 et E2 sont aléatoires (utiliser par ex. le générateur RND() de vBnet) entre deux valeurs min et max.

-Les sorties sont aussi aléatoires, au nombre de DEUX S1 et S2.

Pour simplifier la situation, on tirera les arrivées entre 1 et 9. Si c'est la valeur 3, alors, il y a une arrivée à l'entrée E1 du parking. Si c'est 7, il y a eu une arrivée par l'entrée E2. Sinon, il n'y a pas d'arrivée dans ce cycle. Le cycle sera un intervalle géré par un timer, toutes les 5 secondes par ex on teste l'arrivée d'un client. La fonction est :

`randomize()`

`arrivee = CInt(Math.Floor((9 - 1 + 1) * Rnd())) + 1`

Pour ce qui est des sorties, on peut aussi utiliser un générateur et un timer. On tire un nombre entre 1 et 9. Si c'est 1, il y a une sortie par la sortie S1, si c'est 2, il y a une sortie par la sortie S2 sinon, il n'y a pas de sortie.

Bilan d'un cycle

On veut *connaître les arrivées par E1 (NE1) et E2 (NE2), les sorties par les différentes portes NS1 et NS2, au bout d'un certain nombre de cycles* (1 cycle = 5 s par ex).

A)-Développez un programme qui est une simulation et doit permettre de **visualiser en continu** les nombres totaux d'arrivées, de sorties et surtout le nombre de **places libres**.

Un bouton permettant **d'interrompre la simulation, sans l'arrêter** définitivement, est souhaité.

B)-Développez une variante en tenant compte du fait qu'il y a à l'intérieur DEUX ZONES pour se garer Z1, côté gauche et Z2 côté droit. L'affichage doit aider l'utilisateur qui arrive à savoir quelle zone a des places libres.

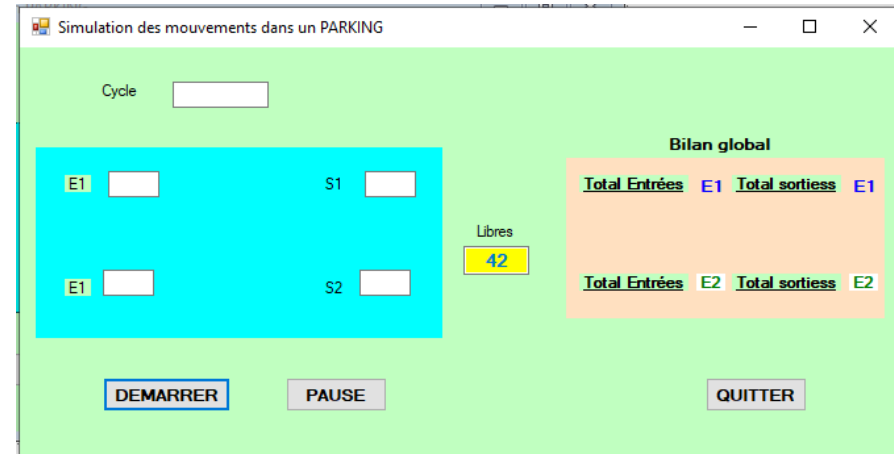
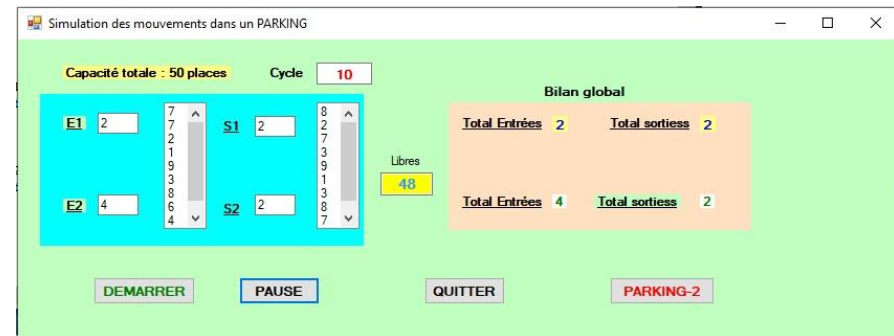


Fig1 : simulation des arrivées et sorties cas simple



Coup de pouce VBnet

Pour les développeurs sous VBnet, il est conseillé d'utiliser TROIS objets TIMER. Le premier pour la gestion des cycles, le second pour le tirage des arrivées et le troisième pour simuler les sorties du parking. On peut considérer un nouveau cycle toutes les 5 secondes dans le premier Timer.

Pour tester les arrivées, on tire un nombre entre 1 et 9 ;

-si on a {3,9}, il y a une arrivée par l'entrée E1, donc NE1 doit être incrémenté.

-si on a {2,7}, il y a une arrivée par l'entrée E2.

Sinon, il n'y a pas d'arrivée.

Pour tester les sorties, on tire un nombre entre 1 et 9 ;

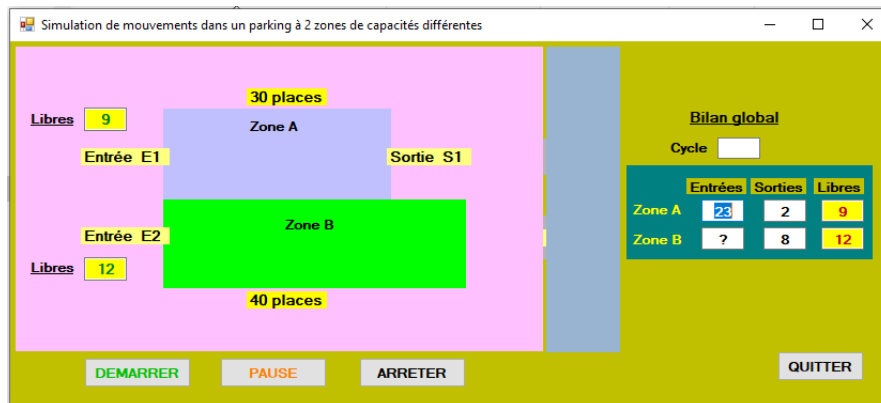
-si on a {4,8}, il y a une sortie par S1.

-si on a {1,5}, il y a une sortie par S2.

Sinon, il n'y a pas de sortie.

Proposition d'une interface de simulation

-pour la variante avec deux entrées et deux ou trois zones de parking ZA, ZB ou ZC.



** Vous pouvez modifier ces propositions d'interfaces. L'important, ce sont les résultats attendus dans les bilans globaux.