

Examen du 17 décembre 2013

Université Paris Diderot

Barème indicatif. Documents interdits. Toute la programmation se fait en SCADE
Durée : 3h (de 15h30 à 18h30). Salle : 418C.

Exercice 1 (3 pts) – Terminologie

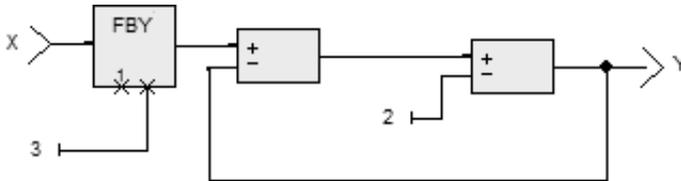
Expliquez en 1-3 lignes chacune des notions suivantes. Barème avec bonus-malus : réponse fausse -0.25 ; pas de réponse 0 ; réponse incomplète ou imprécise 0,25 ; bonne réponse 0,5.

1. Certification
2. Hypothèse synchrone
3. Logiciel critique
4. Langage à flot de données
5. Validation par revue de code
6. Logiciel de temps réel dur.

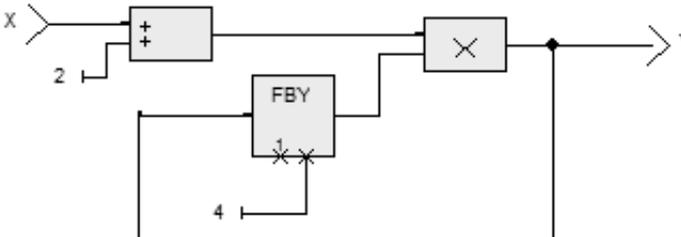
Exercice 2 (2pts) – Modèle synchrone

Est-ce que les opérateurs Scade ci-dessous sont corrects ? Si c'est le cas quelle est leur sortie Y pour l'entrée $X = 1, 2, 3, 4, 5, 1, \dots$ Sinon expliquez pourquoi.

1. Noeud A



2. Noeud B



Exercice 3 (2 pts) – Vérification

Étant donné un noeud tralala avec les entrées entières m, n et une sortie booléenne *alarme*, comment peut-on démontrer avec Scade chacune des propriétés suivantes :

1. si $m > n$ pendant 10 ticks, alors l'alarme sonne ;
2. si toujours $m > n$, alors à chaque instant où $m = 15$ l'alarme sonne.

Exercice 4 (3 pts) – gaz carbonique et ordre supérieur

Dans 20 salles d'un bâtiment on mesure la concentration de CO₂, les résultats constituent une entrée un tableau de 20 réels `co2`.

1. Programmer un noeud qui prend en entrée `co2: real^20` et qui produit une sortie `alerte: bool^20` telle que `alerte[i]=true` si et seulement si `co2[i]>0.05` (c-à-d `alerte[i]` est vrai si dans la salle `i` il y a trop de gaz carbonique).
2. Programmer un noeud qui calcule à tout instant le nombre `nbAlertes` de salles qui sont en état d'alerte (c-à-d le nombre de `i` tels que `alerte[i]=true` à l'instant courant).
3. Programmer un noeud qui calcule le nombre `nbAlertesMoyen` qui est la moyenne du `nbAlertes` durant tout le fonctionnement du système.

Exercice 5 (8pts) – Pancréas artificiel

Dans cet exercice vous devez programmer un pancréas artificiel qui aide à réguler le taux de sucre dans le sang d'un malade de diabète, ce qui se fait en mesurant ce taux et en injectant deux produits : insuline qui fait baisser le sucre, et glucagon qui le fait augmenter.

Votre programme doit représenter un système complet. Faites attention à la clarté et la modularité de votre code. Votre système aura

- l'entrée réelle `suc` qui correspond au capteur qui mesure la quantité du sucre dans le sang ;
- les sorties `ins` and `glu` qui correspondent aux quantité d'insuline et glucagone que le système injecte dans le sang ;
- quelques entrées booléennes correspondant aux boutons de commande et quelques sorties booléennes pour les voyants et les alarmes - à vous de choisir.

Un `tick` correspond à une minute.

1. Programmez d'abord le système très basique qui à chaque `tick` injecte la quantité d'insuline selon la formule

$$\text{ins} = \begin{cases} 3 * (\text{suc} - 90), & \text{si } \text{suc} > 120 \\ 0, & \text{sinon.} \end{cases} \quad (1)$$

et un autre pour le glucagon :

$$\text{glu} = \begin{cases} 4 * (90 - \text{suc}), & \text{si } \text{suc} < 50 \\ 0, & \text{sinon.} \end{cases} \quad (2)$$

2. modifiez le système pour que
 - le sous-système (1) fonctionne en permanence ;
 - le sous-système (2) s'active si pendant 6 minute le taux de sucre est bas (< 48) et se désactive si pendant 20 minute il est > 57 .
3. Ajoutez une alarme qui s'active si pendant 10 minutes le taux de sucre est très bas (< 40) ou très haut (> 190). Elle est représentée par une variable booléenne `alarme`.
4. Ajoutez quelques boutons de commande et un programme qui permettent d'arrêter et réactiver indépendamment les trois fonctionnalités de l'appareil : injection d'insuline, injection de glucagon, alarme.
5. Ajoutez un sous-système qui calcule et affiche la quantité totale de chacun des deux produits injectés depuis le lancement du système. Prévoyez un bouton `reset` pour remettre à zéro ces quantités affichées.
6. comment vérifier la propriété

Il est impossible que le système injecte à la fois les deux produits (au même `tick`).