

DS N° 1 INFORMATIQUE I

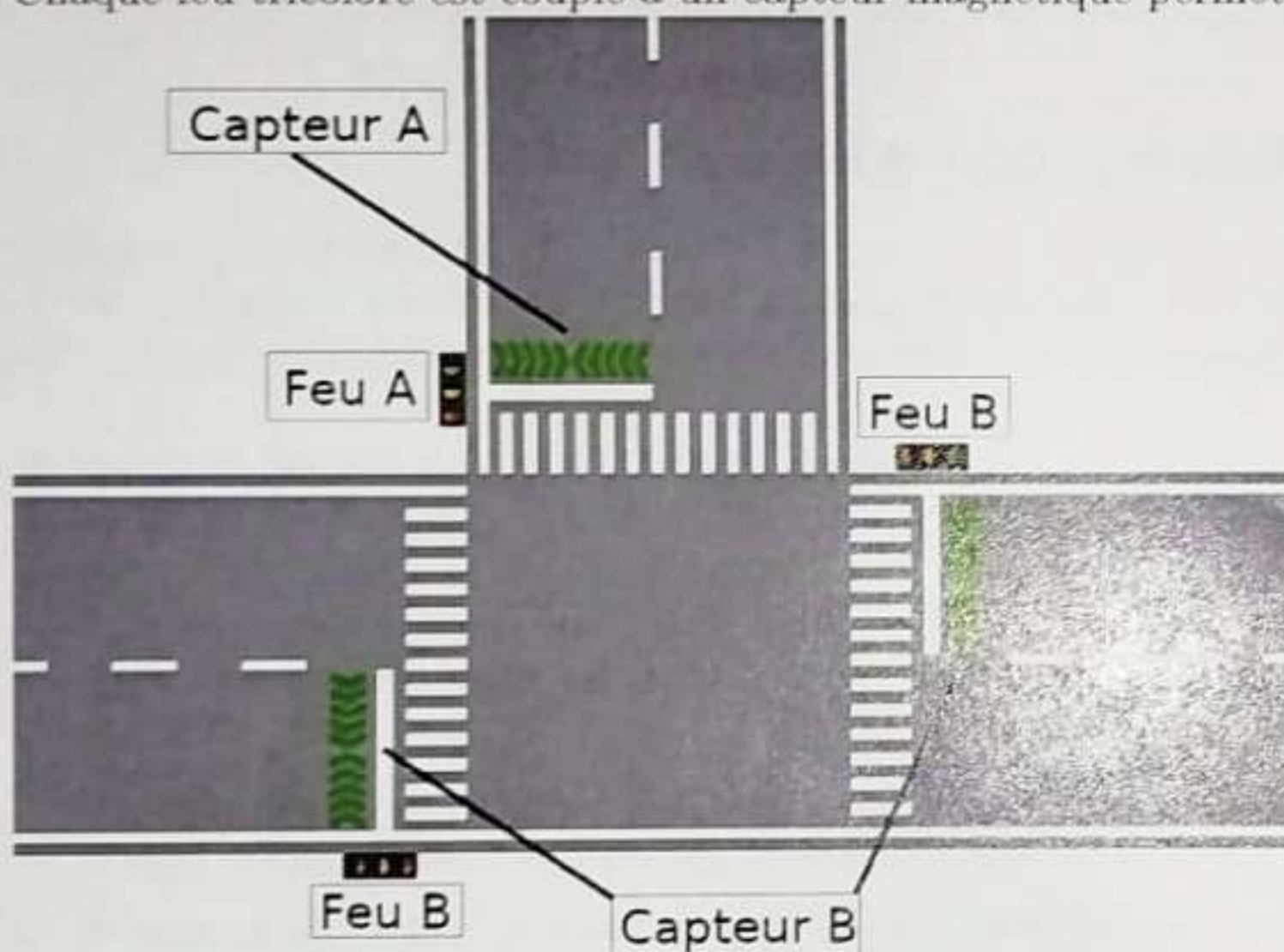
Calculatrice et documents non autorisés

Le langage utilisé peut-être au choix le pseudo code ou le langage C (sauf consigne spécifique).  
Le barème est donné à titre indicatif mais peut être sujet à changement.

Exercice 1 (*Gestion de feux tricolores*) (6.5 pts)

Le but de cet exercice est de concevoir un circuit logique permettant de piloter des feux tricolores d'un carrefour routier en fonction de la présence de voitures. Dans cet exercice, on simplifiera volontairement le fonctionnement, et nous ne tiendrons pas compte des délais, de changement d'état des feux tricolores, ni des temps d'attente des voitures aux feux rouges.

Le carrefour est constitué d'une grande route B, et perpendiculairement à cette dernière, d'une petite route A. Ce carrefour est équipé de 3 feux tricolores, deux pour chaque côté de la route B, et 1 pour la route A. Chaque feu tricolore est couplé à un capteur magnétique permettant de détecter la présence d'un véhicule.

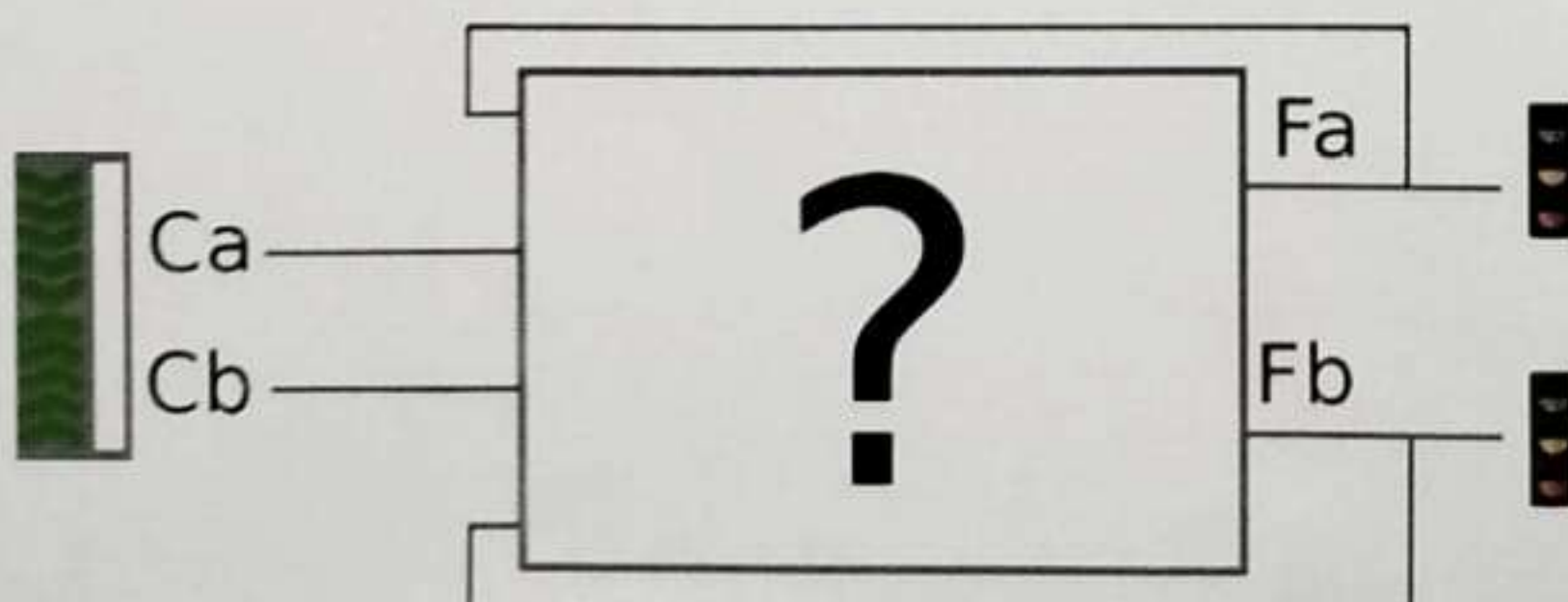


Les 2 capteurs B sont considérés comme un seul et unique capteur.  
Les 2 feux B sont considérés comme un seul et même feu tricolore.

Les dénominations des différents éléments sont les suivantes :

- feu tricolore de la route A : 'Fa'
- feu tricolore de la route B : 'Fb'
- capteur de la route A : 'Ca'
- capteur de la route B : 'Cb'

Le schéma fonctionnel du circuit est constitué de 2 entrées (capteurs Ca et Cb), de 2 sorties (les feux Fa et Fb), mais il va falloir rajouter comme entrées supplémentaires les états des 2 feux eux-mêmes. En effet en l'absence de voitures, les sorties des feux doivent conserver leurs états et ne pas changer.



Les valeurs de Fa et Fb sont à '1' (activé) quand les feux sont au vert (laissant passer les voitures).  
Les valeurs de Ca et Cb sont à '1' (activé) quand les capteurs sont actifs (les voitures attendent pour passer).

Détail fonctionnel attendu :

- Lorsqu'un feu va être activé (passe au vert), l'autre doit obligatoirement être désactivé (passer au rouge)
- Lorsque le capteur Ca est activé, et que le capteur Cb est désactivé, le feu Fa passe au vert.
- Lorsque le capteur Cb est activé, et que le capteur Ca est désactivé, le feu Fb passe au vert.
- Lorsqu'aucun capteur n'est activé (aucune voiture au carrefour), l'état des feux Fa et Fb reste inchangé.
- Lorsque les 2 capteurs sont activés (des voitures sont présentes sur les 2 routes), si les 2 feux sont désactivés, le feu Fb doit s'activer, sinon l'état des feux Fa et Fb reste inchangé.
- Les règles précédentes ne sont plus applicables si les 2 feux Fa et Fb venaient à être activés en même temps par un bug indépendant de notre volonté. Dans ce cas précis, les 2 feux doivent être désactivés.  
En d'autres termes, si l'état précédent des 2 feux est au vert, Le prochaine état de sortie des 2 feux doit être de passer au rouge, et ce, quel que soit l'état des 2 capteurs Ca et Cb.

1. Créer la table de vérité du circuit logique demandé
2. Dessiner le tableau de Karnaugh et en déduire les équations simplifiées des 2 sorties.
3. Dessiner le circuit logique d'après les équations précédentes.

## Exercice 2 (Correction et analyse de code) (3 pts)

1. Le code suivant comporte des erreurs sur 6 lignes. Indiquer les lignes et les corriger (en C).

```
1      #include <studio.h>
2
3      int main() {
4          int a;
5          float c;
6          printf("Saisir un entier ");
7          scanf("%d", &a);
8          printf("Saisir un reel");
9          scanf("%d", &c);
10         b=15;
11         if(a <= c){
12             printf("Chocolat");
13             if (a = c && a < b){
14                 printf("Caramel")
15             }
16             else if(a > b){
17                 printf("Truffe");
18             }
19             else(a>c){
20                 printf("Bonbons");
21             }
22         printf("Glaces");
23         return 0;
24     }
```

### Exercice 3 (*Devises*) (5 pts)

Le but de cet exercice est de réaliser un convertisseur depuis l'Euro (€) vers d'autres devises internationales.

1. Le programme doit afficher les devises et demander à l'utilisateur d'entrer une valeur entière pour choisir la monnaie de destination :

- valeur 1 : pour convertir en Won Sud-Coréen (W).
- valeur 2 : pour convertir en Peso Argentin (\$)
- valeur 3 : pour convertir en Forint Hongrois (Ft)

Toute autre valeur entrée par l'utilisateur doit générer l'affichage d'un message d'erreur pour indiquer que le programme ne peut pas gérer d'autres devises, et le programme doit se stopper.

2. Si le choix de la devise est correct, le programme doit ensuite demander à l'utilisateur d'entrer le montant en Euros (nombre réel).

Si le montant en Euros est négatif ou nul, un message d'erreur sera affiché pour indiquer qu'il n'y a pas de montant à convertir, et le programme doit se stopper.

3. Si la valeur en Euro est correcte, le programme doit effectuer le calcul de conversion vers la devise demandée.

- 1 Euro vaut 1473.92 Won
- 1 Euro vaut 1067.04 Pesos
- 1 Euro vaut 398.94 Forint

Le programme affichera le résultat du calcul de conversion ainsi que la devise choisie. Exemple : si l'utilisateur a choisi de convertir en Won et qu'il a entré 3.5 Euros comme valeur, le résultat affiché pourra être similaire à :

3.5€ <=> 5158.72 Won

### Exercice 4 (*Lancé de Dés*) (5,5 pts)

En C la commande `rand()` (qui s'écrit `ALEA()` en pseudo-code) permet d'obtenir n'importe quel entier naturel aléatoire (positif ou nul). (Attention, cette fonction ne prend pas d'argument!).

Nous allons simuler un jeu de paris utilisant des dés.

Voici les règles du jeu :

- Un joueur mise une somme d'argent.
- Il peut choisir entre lancer un dé, deux dés ou 3 dés.
- Sa mise sera multipliée par la somme des dés MAIS il perdra si jamais deux dés lancés sont égaux et devra alors payer 20 fois sa mise.

1. Quelle formule pourrait-on appliquer pour simuler le résultat d'un lancé de dés ?

*Si vous ne savez pas répondre, on remplacera par "instruction\_lancer" cette commande pour le reste de l'exercice.*

2. Demander au joueur de saisir sa mise et la stocker puis lui demander le nombre de dés avec lesquels il souhaite jouer.

3. Si jamais la mise est négative ou nulle, ou si le nombre de dés voulu n'est pas valide, afficher un message d'erreur et stopper le programme, sinon afficher si le joueur a gagné ou perdu de l'argent ainsi que la valeur gagnée ou perdue.