

CONCOURS INTERNE
POUR LE RECRUTEMENT DE CONTRÔLEURS DES DOUANES ET DROITS INDIRECTS

BRANCHE DU CONTRÔLE DES OPÉRATIONS COMMERCIALES ET D'ADMINISTRATION GÉNÉRALE

DES 23 ET 24 FÉVRIER 2015

ÉPREUVE ÉCRITE D'ADMISSIBILITÉ N°2

(DURÉE : 5 HEURES - COEFFICIENT 5)

OPTION D : INFORMATIQUE

PUPITREUR : page 2

PROGRAMMEUR : pages 3 à 6

AVERTISSEMENTS IMPORTANTS

Si, lors de votre inscription, vous avez **choisi d'obtenir la qualification de pupitreur**, vous devez impérativement composer dans le seul sujet permettant d'obtenir cette qualification sinon votre copie sera notée zéro.

Si, lors de votre inscription, vous avez **choisi d'obtenir la qualification de programmeur**, vous devez impérativement composer dans le seul sujet permettant d'obtenir cette qualification sinon votre copie sera notée zéro.

Veillez à bien indiquer sur votre copie **la qualification** pour laquelle vous allez composer ainsi que le **nombre d'intercalaires** utilisés (la copie double ne compte pas).

A l'exception de l'organigraphe, l'usage de tout matériel autre que celui d'écriture et de tout document autre que le support fourni est **interdit**.

Toute fraude ou tentative de fraude constatée par la commission de surveillance entraînera **l'exclusion du concours**.

Il vous est interdit de quitter définitivement la salle d'examen **avant le terme de la première heure**.

Le présent document comporte **6 pages** numérotées.

SUJET POUR OBTENIR LA QUALIFICATION DE PUPITREUR

Remarques préliminaires :

- *Toutes les questions devront être traitées. Chaque réponse devra être précédée du numéro de la question à laquelle elle se rapporte.*
- *Toutes les réponses devront être justifiées.*
- *Tous les schémas et diagrammes éventuellement réalisés devront être accompagnés d'un commentaire expliquant la valeur de leurs symboles.*

Question n° 1 :

Quel est l'intérêt d'utiliser la signature électronique ? Vous préciserez de quelle manière elle est mise en œuvre.

Question n° 2 :

Quelles sont les différentes phases de la vie d'un virus ? Expliquez ce qu'est un virus système.

Question n° 3 :

Comment est organisée une chaîne d'assistance aux utilisateurs ? Vous présenterez également les missions d'un assistant utilisateurs.

Question n° 4 :

Le « travail collaboratif » : Expliquez ce qu'on entend par « collaboratif » dans le domaine de l'informatique en présentant les différentes techniques utilisées.

Question n° 5 :

Une salle de formation dotée d'ordinateurs accueille régulièrement des stagiaires. Quel plan de sauvegarde peut être mis en œuvre pour protéger l'ensemble des données stockées sur les postes de travail et comment rétablir le système pour une nouvelle session ?

SUJET POUR OBTENIR LA QUALIFICATION DE PROGRAMMEUR

Remarques préliminaires :

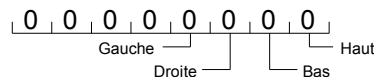
- *Vous devez traiter le sujet dans le langage choisi lors de votre inscription. Si vous traitez le sujet dans un langage différent, votre copie sera notée zéro.*
- *Vous préciserez au début de votre devoir le langage choisi.*
- *Toutes les questions devront être traitées. Chaque réponse devra être précédée du numéro de la question à laquelle elle se rapporte.*
- *Toutes les réponses devront être justifiées.*
- *Si des options vous semblent nécessaires après la lecture de tous les éléments fournis, il vous appartiendra de les indiquer et de les justifier.*
- *Tous les schémas et diagrammes à réaliser devront être accompagnés d'un commentaire expliquant la valeur de leurs symboles.*

Vous participez au développement d'un drone chasseur de drone (DCD). Le DCD est un mini-drone économique capable de générer une impulsion électromagnétique (IEM) grillant les composants électroniques situés à un mètre de lui. Votre rôle est d'écrire son programme de pilotage et d'activation de l'IEM.

De votre point de vue, le drone est composé de trois éléments : un système de déplacement, un système de détection et un système d'activation du champ électromagnétique.

Le système de déplacement est contrôlé par les 4 premiers bits d'une variable `dep` de type entier non signé d'une taille de 8 bits.

Figure 1 : structure de `dep`



La structure de `dep` est illustrée par la Figure 1. Lorsque le bit 0 est activé (mise à 1), le drone monte. Lorsque le bit 1 est activé, il descend. Les bits 2 et 3 font respectivement pivoter le drone à droite et à gauche. Dans le cas où les bits 4, 5, 6 et 7 sont activés, ou dans le cas où les bits 0 et 1 sont actifs simultanément, ou dans le cas où les bits 2 et 3 sont actifs simultanément, une sécurité du système de déplacement active l'IEM provoquant l'autodestruction du DCD. Le drone étant conçu comme un missile, on considère que le DCD avance en continu vers l'avant.

Le champ électromagnétique est contrôlé par une variable `emp` de type booléen. Lorsque la valeur vraie est écrite dans la variable, le système s'active.

Le système de détection est composé d'un fichier `/dev/scan` contenant un tableau à taille variable d'objets identifiés. Les éléments du tableau sont de type `Obj` qui est une structure contenant un vecteur et une énumération de type `TypObj` indiquant le type d'objet détecté. `TypObj` utilise la valeur `drone_ennemi` pour indiquer un drone à détruire.

Figure 2 : Représentation algorithmique du type `Obj`

```
Type Obj : Réel :      x
           Réel :      y
           Réel :      z
           TypObj :    type
```

Le vecteur a trois coordonnées `x`, `y` et `z` de type réel.

`x` indique à quelle distance (en mètres) l'objet se trouve devant (si positif) ou derrière (si négatif) le drone.

`y` indique à quelle distance (en mètres) l'objet se trouve à droite (si positif) ou à gauche du drone.

`z` indique à quelle altitude (en mètres) l'objet se trouve par rapport au DCD.

Dans ce référentiel, le DCD a toujours les coordonnées (0, 0, 0). Les trois coordonnées des objets sont calculées par rapport à la position du drone.

Contrôle de l'IEM

1) Ecrivez l'algorithme en pseudo-code d'un sous-programme `boom` activant l'IEM.

Contrôle de la direction

2) Ecrivez l'algorithme en pseudo-code d'un sous-programme `set_dep` qui prend un entier 8 bits et qui l'écrit dans la variable `dep` après avoir vérifié que la valeur fournie n'entraînera pas l'autodestruction du drone.

3) Ecrivez l'algorithme en pseudo-code d'un sous-programme `changer_direction` à deux arguments. Le premier argument indique si le drone doit tourner à gauche ou à droite. Le deuxième indique si le drone doit prendre ou perdre de l'altitude. Le sous-programme `changer_direction` définit la variable `dep` pour obéir à l'ordre fourni par les deux arguments.

Expliquez brièvement la pertinence de votre choix concernant le type choisi pour les deux arguments.

Analyse de l'environnement

4) Ecrivez l'algorithme en pseudo-code d'un sous-programme `distance` qui prend en argument un objet identifié de type `Obj` et qui calcule la distance séparant le drone de l'objet.

La distance se calcule avec la racine carrée de la somme de `x` au carré plus `y` au carré plus `z` au carré.

5) Ecrivez l'algorithme en pseudo-code d'un sous-programme `charger_environnement` qui charge en mémoire dans un tableau le contenu du fichier `/dev/scan`.

6) Ecrivez l'algorithme en pseudo-code d'un sous-programme `filtrer_obj` qui prend un argument `selecteur` de type `TypObj` et un argument `liste` de type tableau d'`Obj` ainsi que sa taille et qui retourne la liste des index de tous les objets identifiés dont le type est égal au `selecteur`.

Expliquez en une ou deux lignes comment est gérée la taille variable du tableau d'index.

7) Après avoir décrit un algorithme de tri simple à implémenter mais peu efficace, créez un sous-programme `trier_obj` qui prend en argument un tableau d'`Obj` et qui retourne la liste des index de tous les objets identifiés triés par ordre croissant de distance les séparant du drone.

8) Supposons que la liste d'objets identifiés contient plus d'un million d'éléments. Quel type de structure de données permettrait d'indexer efficacement les objets identifiés en fonction de leur distance ? Expliquez en une quinzaine de lignes le fonctionnement de cette structure ainsi que les complexités algorithmiques de l'ajout et de la recherche d'un élément. L'implémentation en pseudo-code de cette structure de données est exclue de l'exercice.

Détonation sécurisée

9) Ecrivez l'algorithme en pseudo-code d'un sous-programme `activer_iem` qui appelle le sous-programme `boom` si et seulement s'il n'y a pas d'objet à moins de trois mètres du DCD, autre que des drones ennemis.

Programme principal

10) Ecrivez l'algorithme en pseudo-code d'un programme `activer` qui dirige constamment le DCD vers le drone ennemi le plus proche et qui active l'IEM au bon moment. Votre programme devra utiliser les sous-programmes définis dans l'exercice.

Créez à bon escient un ensemble de sous-programmes pour la réalisation du programme `activer`.

11) Tel que l'énoncé de l'exercice le présente, la recherche du drone ennemi le plus proche n'est pas efficace pour un grand nombre d'objets identifiés. Il est possible d'améliorer de manière drastique l'efficacité du programme en ajoutant un paramètre à un des sous-programmes. Indiquez le paramètre manquant, ainsi que les gains attendus en termes de temps d'exécution du sous-programme. L'implémentation de cette optimisation est exclue de l'exercice.

Variable de contrôle

12) Les variables `dep` et `emp` sont directement liées au matériel. Il n'y pas de driver logiciel traduisant le contenu des variables en signaux électriques contrôlant le matériel. Expliquez brièvement pourquoi c'est possible.

Codage

13) Implémentez l'ensemble des algorithmes de l'exercice avec le langage de programmation sélectionné. Concernant l'implémentation, vous êtes libre de faire preuve d'adaptation lorsque le langage de programmation ne vous fournit pas les outils nécessaires. Par exemple, un espace de stockage peut être utilisé pour implémenter des tableaux ou les variables `dep` et `emp`. Dans le pire des cas, vous êtes autorisés à inventer de nouvelles instructions à condition de les documenter.
