

Si LICENCE SCIENCES POUR L'INGENIEUR – L3 – SESSION 1  
EXAMEN DE S61 – PROGRAMMATION III – C++  
Durée 2h – Documents non autorisés – Mai 2016

Cet exercice s'intéresse, de façon très simplifiée, à la location de gros outillage.

Les outils sont classés par catégorie (NettoyeurHP, Débroussailleuse, Bétonnière, etc...).

Un outil d'une catégorie donnée a un prix de location par jour.

Pour un outil d'une catégorie donnée, il y a plusieurs exemplaires de cet outil susceptibles d'être loués par le magasin, chacun est identifié par un code et on mémorise sa disponibilité. Ces codes sont attribués par le magasin de location, ils sont tous distincts et chacun des exemplaires a son code apparent.

**Exemple1** : Un outil qui est un NettoyeurHP, loué 18 Euros par jour et pour lequel il y a 5 exemplaires susceptibles d'être loués, celui de code "24ST280" qui a été loué et n'est donc pas disponible, ceux de codes "357HY400", "293ST350", "125PE600" et "539HO78" qui sont disponibles.

Il est recommandé de commencer par lire attentivement le sujet jusqu'à la fin.

1) Ecrire le fichier d'extension h et celui d'extension cc de la classe **Outil** définie ci-dessous et indiquer pourquoi elle ne possède pas de destructeur.

Les données membres sont : **categorie** de type string et **prix** (prix d'un jour de location de l'outil) de type float.

**Exemple2** : Dans le cas de l'**Exemple1**, l'attribut **categorie** vaut "NettoyeurHP", l'attribut **prix** vaut 18.

Les fonctions membres sont : un constructeur sans argument, un constructeur à 2 arguments, un constructeur de recopie, 2 accesseurs **Getcategorie** et **Getprix**, un modificateur **Modprix** et une fonction **Affic** d'affichage.

2) Ecrire le fichier d'extension h et le fichier d'extension cc de la classe **Alouer** ainsi définie :

Les données membres sont : **code** de type string et **dispo** qui est un booléen.

Les fonctions membres sont :

- un constructeur sans argument
- un constructeur qui a un argument de type string et qui met **dispo** à vrai
- 2 accesseurs **Getcode** et **Getdispo**
- un modificateur **Moddispo**

L'affichage d'une instance de **Alouer** est fait en surchargeant l'opérateur d'écriture <<.

3) Ecrire le fichier d'extension h et le fichier d'extension cc de la classe **Outilalouer** ainsi définie :

Elle hérite de la classe **Outil** avec, en outre, les données membres :

- **n** qui, pour un outil d'une catégorie donnée, désigne le nombre d'exemplaires, de cet outil, susceptibles d'être loués
- **u** le vecteur dynamique de taille **n** d'instances de la classe **Alouer** qui, pour un outil d'une catégorie donnée, mémorise les exemplaires, susceptibles d'être loués, de cet outil

**Exemple3** : Dans le cas de l'**Exemple1**, **categorie** vaut "NettoyeurHP", **prix** vaut 18, **n** vaut 5, les éléments de **u** ont pour codes respectifs "24ST280" (pour lequel **dispo** est faux), "357HY400", "293ST350", "125PE600" et "539HO78" (pour lesquels **dispo** est vrai).

Les fonctions membres sont :

- un constructeur sans argument, un constructeur à 3 arguments (la catégorie, le prix, le nombre d'exemplaires), un constructeur à 4 arguments et un constructeur de recopie
- un destructeur
- un accesseur **Getn**
- une surcharge de l'opérateur **[]** qui, à partir de l'objet courant, permet d'accéder à un élément de **u** d'indice donné
- une fonction **Affic** qui affiche toutes les données d'une instance d'**Outilalouer**
- une fonction **Chercher** qui prend un argument **c** de type string et retourne l'indice dans **u** de l'exemplaire de code **c** (on suppose qu'il existe dans **u** un tel exemplaire)
- une fonction **Chercherdisponible** qui retourne l'indice dans **u** du premier exemplaire disponible s'il en existe un et -1 sinon
- une fonction **Louer** qui prend en argument un entier **i** et réalise la location de l'exemplaire d'indice **i** dans **u** (on suppose que cette location est possible)
- une fonction **Restituer** qui prend un argument **c** de type string et réalise la restitution de l'exemplaire de code **c** (on suppose que cet exemplaire a été loué)

**Ce n'est pas terminé, il faut tourner la page!**

- une surcharge de l'opérateur d'affectation
- une fonction *Enlever* qui prend un argument *c* de type string et retourne l'instance d'*Outilalouer* obtenue, à partir de l'objet courant, en retirant de *u* l'exemplaire de code *c* (on supposant que cet exemplaire figure dans *u* et qu'il est disponible)
- une surcharge de l'opérateur + qui retourne l'instance d'*Outilalouer* obtenue, à partir de l'objet courant, en ajoutant un nouvel exemplaire.

#### 4) Programme de test

- a) Ecrire un programme qui crée et affiche l'instance *z* de *Outilalouer* qui est un "NettoyeurHP", loué 18 Euros par jour et pour lequel il y a 5 exemplaires qui sont tous disponibles et ont pour codes "24ST280", "357HY400", "293ST350", "125PE600" et "539HO78".
- b) On suppose que l'instance *z* de *Outilalouer* décrite dans le 4) a) a été créée et qu'un certain nombre de locations et de restitutions portant sur *z* ont été faites. Ecrire les lignes de programme qui permettent de :
- chercher le premier exemplaire disponible
  - demander le nombre de jours de location
  - louer cet exemplaire et afficher le montant de la location
  - restituer un exemplaire dont le code sera lu (on suppose que cet exemplaire a été loué)
  - enlever un exemplaire dont le code sera lu (on suppose que cet exemplaire figure dans *u* et qu'il est disponible)

## QUESTIONS DE COURS

### Classe générique

- 1) Donner la déclaration de la classe générique *Duo* dont les données membres sont *id* de type T et *val* de type float et qui possède (entre autres) un constructeur à 2 arguments, un constructeur de copie et une fonction d'affichage.
- 2) Implanter le constructeur de copie.
- 3) Ecrire un programme qui
  - crée *d1* instance de *Duo* pour laquelle *id* vaut "ananas" et *val* vaut 2.5
  - crée une copie *d2* de *d1*
  - affiche *d2*

### Exception

- 1) Ecrire une fonction *Calcul* qui prend 2 arguments *a* et *b* flottants et qui retourne *a/b* si *b* est non nul et qui déclenche une exception *Erdenom* sinon.
- 2) Ecrire un programme qui lit 2 nombres *x* et *y* puis affiche le résultat de *Calcul* si *y* n'est pas nul et rattrape l'exception déclenchée par *Calcul* sinon en affichant le message d'erreur "Mauvais dénominateur".

# 162, Interface homme-machine

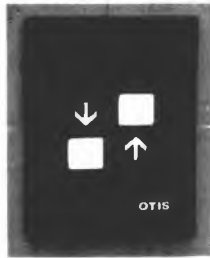
L3

30 juin 2016

La précision et la clarté de votre rédaction sont *fondamentales*. Chaque réponse doit être accompagnée d'une *justification*, dans le cas contraire elle sera purement et simplement ignorée. Le barème est donné à titre indicatif. Durée 2 h.

## Exercice 1. [8 pts] Conception

- [2 pts] Expliquer en terme d'ergonomie (utilisabilité) les mérites respectifs de la manipulation directe (souvent réalisée à la souris) et des langages de commandes. Pour cela, citer au moins deux intérêts et deux inconvénients (critères d'ergonomie vérifiés et transgressés) pour la manipulation directe et pour les langages de commandes. Justifier votre réponse.
- [2 pts] La photo ci-dessous représente un panneau de commande d'ascenseur. Donner deux critères d'ergonomie non respectés. Justifier votre réponse.



- [2 pts] Il y a un an, la SNCF a changé son affichage sur les quais franciliens. Donner deux avantages (critères d'ergonomie vérifiés) de cette nouvelle présentation des informations. Justifier votre réponse.

AVANT



APRÈS



- [2 pts] Pour une application donnée, le fait de quitter sans sauvegarder affiche une boîte de dialogue dans laquelle est inscrit :
  - soit le message : « Voulez-vous vraiment quitter sans avoir sauvé? »,
  - soit le message : « Voulez-vous sauver avant de quitter? ».
 Quelle solution adopter de préférence? Justifier votre réponse.

## Exercice 2. [12 pts] Conception d'un équivalent électronique aux "Post-It", des rectangles de papier partiellement adhésifs qui sont utilisés comme pense-bêtes.

- [3,5 pts] Conception ergonomique
 

Le système retenu doit offrir les fonctionnalités suivantes : création, édition, impression et destruction d'un "Post-It", sauvegarde automatique, association d'une alarme en option avec chaque "Post-It", recherche d'un "Post-It" et édition des préférences.

Pour chaque principe ergonomique associé aux interfaces graphiques (manipulation directe, cohérence, concision, retour d'informations, structuration des activités, flexibilité et gestion des erreurs), donner un exemple associé à cette application particulière.
- [4 pts] La fenêtre "Post-It"
 

Un "Post-It" se présente sous la forme d'une fenêtre qui contient une zone éditable. Un bouton en haut à gauche permet d'ouvrir un menu associé à la note. Ce menu permet de changer la couleur du "Post-It", sa police de caractère, de lui associer une alarme, de le détruire ou de l'imprimer. Lorsqu'une alarme est spécifiée, un réveil est affiché en haut à droite de la fenêtre du "Post-It" correspondant. L'utilisateur doit avoir la possibilité d'icônifier la note.

Avec l'aide de la bibliothèque tkinter, donner le code Python permettant la création et l'affichage de cette fenêtre (sans implanter le code correspondant aux options du menu).
- [4,5 pts] Associer une alarme à un "Post-It"
 

Le formulaire de définition d'une alarme doit comporter comme informations non éditables la date de jour et l'heure et doit permettre de spécifier la date et l'heure de l'alarme associée au "Post-It" courant ainsi que sa fréquence d'activation (une seule fois, une fois par jour, une fois par semaine ou une fois par mois) et son activation ou désactivation effective.

Avec l'aide de la bibliothèque tkinter, donner le code Python permettant cette fonctionnalité. Mettre en évidence les structures de données mises en œuvre.

## 162, Interface homme-machine

17 mai 2016

La précision et la clarté de votre rédaction sont *fondamentales*. Chaque réponse doit être accompagnée d'une *justification*, dans le cas contraire elle sera purement et simplement ignorée. Le barème est donné à titre indicatif. Durée 2 h.

### Exercice 1. [6 pts] Interaction et ergonomie

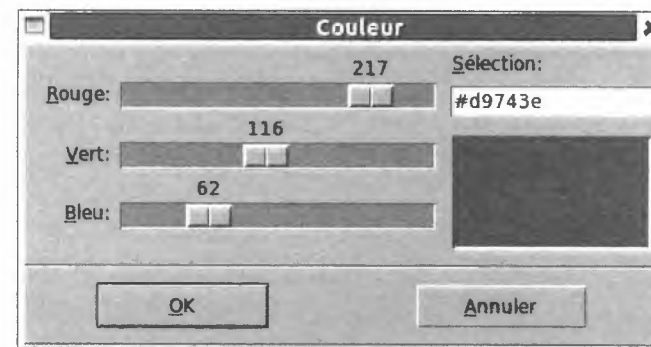
- [1,5 pts] Dans le contexte d'un calendrier numérique, concernant le choix d'une heure on peut utiliser les widgets combobox, entry ou radiobutton. Donner les avantages et les inconvénients de chacun.
- [1,5 pts] En vous reposant sur les critères d'ergonomie (cohérence, concision, structuration des activités, flexibilité, retour d'informations et gestion des erreurs (!)), donner trois règles pratiques de conception à appliquer pour les *messages d'erreur*.
- [3 pts] Pour chacun des styles d'interaction suivant, donner *un* point positif *ou* négatif en le justifiant :
  - (1) langage de commande,
  - (2) interface à base de formulaire,
  - (3) interface à base de question-réponse,
  - (4) interface à base de menu,
  - (5) interface à manipulation directe,
  - (6) interface de réalité virtuelle.

### Exercice 2. [3 pts] Modélisation MVC

Décrire le modèle, la vue et le contrôleur du widget menu.

### Exercice 3. [4 pts] Vue d'une fenêtre de dialogue

- [2 pts] Donner le code Python, qui inclue le module tkinter, permettant la création et l'affichage de la fenêtre de dialogue ci-dessous. On ne demande pas le code permettant la mise en œuvre fonctionnelle de ce dialogue mais uniquement ce qui permet son affichage.
- [2 pts] Identifier chaque composant d'interface de la fenêtre de dialogue ci-dessous qui nécessite la mise en place d'un binding et donner explicitement chaque événement pris en compte. On ne demande pas de coder les bindings.



### Exercice 4. [7 pts] Conception ergonomique

On demande de concevoir la version 2.0 d'un agenda électronique. La conception de la version 1 est donnée et on demande de critiquer et d'améliorer les solutions proposées. Cet agenda est destiné à une population d'utilisateurs allant du novice à l'expert.

#### Analyse de la tâche

- fixer un RDV
  - (1) appeler la commande créer,
  - (2) spécifier le moment du RDV : spécifier un mois puis un jour puis une heure dans la journée.
  - (3) saisir le texte décrivant le RDV.
- supprimer un RDV
  - (1) appeler la commande supprimer,
  - (2) spécifier le moment du RDV : spécifier un mois puis un jour puis une heure dans la journée.
  - (3) valider la suppression.
- consulter un jour
  - (1) appeler la commande consulter
  - (2) spécifier le jour du RDV : spécifier un jour puis un mois.

Critiquer l'analyse de tâche décrit pour la version 1. Pour cela :

- (1) s'appuyer sur des critères d'ergonomie non vérifiés par l'analyse de tâche,
- (2) identifier les concepts (correspondant à des profils d'utilisateurs) non présents dans l'analyse,
- (3) identifier des activités de l'utilisateur impossibles selon cette analyse (s'appuyer pour cela sur un scénario).

Propagation des ondes. Examen 2015-2016 - seconde session. Durée: 2 h

La calculatrice ainsi que les notes de cours sont interdites. Les quantités notées en gras dans les exercices sont des quantités vectorielles. Lorsque ces mêmes quantités vectorielles sont notées sans gras, c'est de leur norme dont il est question. (ex :  $dl = ||dl||$ ). Le symbole  $\times$  signifie le produit vectoriel. Les exercices valent dans l'ordre, 3, 9 et 9 points. Bon courage.

1 Questions de cours

1. Donner les quatre équations de Maxwell dans le vide en présence d'une distribution de charge  $\rho$  et de courants volumiques  $j$ .
2. On rappelle la propriété mathématique :  $\mathbf{rot}(\mathbf{rot} \mathbf{u}) = \mathbf{grad}(\text{div} \mathbf{u}) - \Delta \mathbf{u}$  valide pour  $\mathbf{u}$  quelconque. Retrouver l'équation de propagation dans le vide pour le champ électrique :

$$\Delta \mathbf{E} - \frac{1}{c^2} \frac{\partial^2 \mathbf{E}}{\partial t^2} = \mathbf{grad} \left( \frac{\rho}{\epsilon_0} \right) + \mu_0 \frac{\partial \mathbf{j}}{\partial t}$$

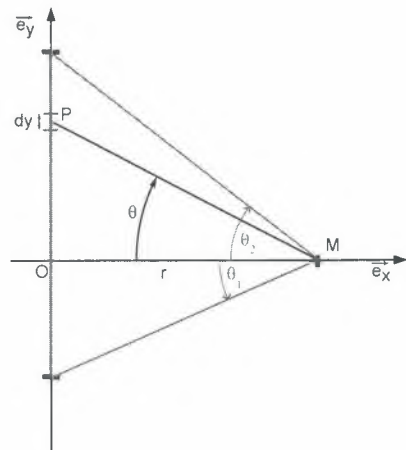
où  $c$  est la vitesse de la lumière dans le vide.

2 Fil fini chargé

On se place dans le repère plan cartésien direct  $(0, \mathbf{e}_x, \mathbf{e}_y)$ . On considère un fil de longueur finie chargé uniformément et possédant une charge linéique constante  $\gamma$ . Il est placé dans le vide à  $x = 0$  et orienté selon l'axe  $\mathbf{e}_y$  comme représenté sur la figure ci-contre.

Les angles  $\theta_1$  et  $\theta_2$  permettent de repérer les deux bouts du fil à partir d'un point  $M$  situé sur l'axe  $\mathbf{e}_x$  à une distance  $r$  du fil.

On se propose de trouver l'expression du champ électrique créé au point  $M(r, 0)$ .



1. Rappeler l'expression générale d'un champ électrostatique créé à une distance  $r$  d'une charge ponctuelle  $q$ . On note  $\epsilon_0$  la permittivité électrique du vide.
2. Donner l'unité de  $\gamma$  ainsi que l'expression liant une petite longueur de fil  $dy$  et la charge  $dq$  qu'il porte.
3. Représenter le champ élémentaire  $d\mathbf{E}_p(M)$  créé en  $M$  par un élément  $dy$  de charge  $dq$  situé au point  $P(0, y)$  du fil (voir le schéma).
4. Donner les expressions des composantes  $dE_x(M)$  et  $dE_y(M)$  de ce champ élémentaire en fonction de  $\epsilon_0, \gamma, PM, dy$  et  $\theta$  où  $\theta$  est l'angle  $(MO, MP)$ .

5. En se rappelant que:

$$\frac{d(\tan(\theta))}{d\theta} = 1 + \tan^2(\theta) = \frac{1}{\cos^2(\theta)}$$

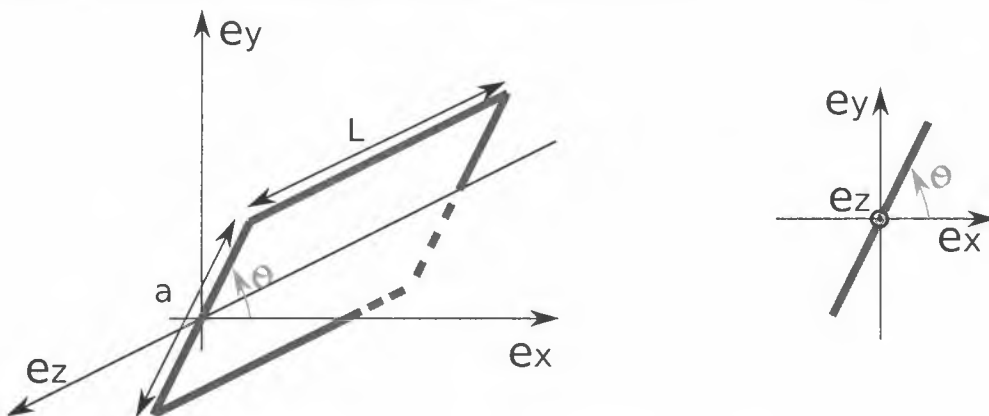
Montrer que l'on peut écrire:

$$dE_x(M) = \frac{\gamma \cos(\theta) d\theta}{4\pi\epsilon_0 r}$$

6. Donner l'expression de la composante selon  $e_x$  du champ électrostatique total :  $E_x(M)$  créé par le fil en fonction de  $\epsilon_0$ ,  $\gamma$ ,  $r$ ,  $\theta_1$  et  $\theta_2$ .
7. Retrouver l'expression du champ électrique créé par un fil infini. Que vaut  $E_y(M)$  dans ce cas et pourquoi ?

### 3 Force électromotrice d'induction

On se place dans le repère cartésien direct  $(0, e_x, e_y, e_z)$  et on considère une spire conductrice rectangulaire représentée ci-après. Cette spire de dimensions  $a \times L$  tourne par son axe milieu autour de l'axe  $e_z$  à une vitesse angulaire constante  $\omega$ . On note  $\theta$  l'angle que fait le plan de la spire avec le plan  $(0, e_x, e_z)$  de telle manière que  $\theta = \omega t$  ( $\theta = 0$  à  $t = 0$ ).



Il règne dans l'espace un champ magnétique  $B$  constant tel que:

$$B = B_0 e_x \quad (0.1)$$

- Rappeler la définition du flux d'un vecteur  $u$  à travers une surface  $S$ .
- Calculer le flux  $\phi$  de  $B$  à travers la spire en fonction de ses dimensions  $a$  et  $L$ , de  $B_0$  et de  $\theta$ . Pour ce faire, on calculera préalablement les composantes selon les axes  $e_x$  et  $e_y$  du vecteur élémentaire de surface  $dS$  de la section de la spire.
- Rappeler la loi d'induction liant la variation temporelle du flux du champ magnétique  $\frac{\partial \phi}{\partial t}$  à travers un circuit fermé et la force électromotrice  $e(t)$  engendrée dans ce circuit fermé.
- En déduire l'expression de la force électromotrice d'induction  $e(t)$  engendrée dans la spire en fonction de  $a$  et  $L$ ,  $B_0$ ,  $\omega$  et  $t$ .
- Un système mécanique associé à la spire permet d'extraire de celle-ci une tension  $U$  telle que  $U(t) = |e(t)|$ . Représenter les formes d'onde de  $e(t)$  et de  $U(t)$  en fonction du temps et calculer la moyenne temporelle de  $U(t)$ .