**SUJET D’EXAMEN GL 40 – 30 juin 2017**

*(documents autorisés)*

**Exercice 1**. Considérons la grammaire suivante (les éléments entre chevrons « $<>$ » indiquent des non terminaux) :

<phrase> $\rightarrow $ <sujet> <verbe> <objet>

<sujet> $\rightarrow $<determinant> <nom>

<objet> $\rightarrow $ <determinant> <nom>

<verbe> $\rightarrow $ mangent | voient

<determinant> $\rightarrow $ les | des

<nom> $\rightarrow $ chats | lions | souris| jambons

Quel est l’axiome ? donner un exemple de dérivation d’une phrase du langage généré par la grammaire en explicitant les règles appliquées.

**Exercice 2**. Considérons les deux langages suivants sur l’alphabet {a, b} :

$$L\_{1}=\left\{\left(ab\right)^{n}/n\geq 0\right\}$$

$$L\_{2}=\left\{a^{n}b^{n}/n\geq 0\right\}$$

Quel est leur type selon la hiérarchie de Chomsky ? Donner une grammaire pour chacun d’eux. Préciser bien axiome et non terminaux. Exprimer le langage sous forme d’expression régulière si possible.

**Exercice 3**. Donner un automate d’état fini reconnaissant le langage défini par l’expression régulière suivante :

$$E=(a+b\left(aba\right)^{\*}aa)^{\*}$$

**Exercice 4.** Considérons le langage défini sur l’alphabet $\left\{0, …, 9,+,-,E,. \right\}$ et généré par la grammaire suivante ($ϵ$ est le mot vide et la barre verticale « | » représente le "ou", $<nb>$ est l’axiome) :

$$<nb>\rightarrow <chiffres><fractionopt><exposantopt>$$

$$<chiffre>\rightarrow 0\left|1\right|2\left|3\right|4\left|5\right|6\left|7\right|8|9$$

$$<chiffres>\rightarrow <chiffre><chiffre>^{\*}$$

$$<fractionopt>\rightarrow . <chiffres> | ϵ$$

$$<exposantopt>\rightarrow \left(E\left(+|-|ϵ\right)<chiffres>\right) | ϵ$$

1) Donner des exemples de mots du langage. A quoi correspond-il de connu ?

2) Donner un automate d’état fini reconnaissant le langage en question.

**Exercice 5.** Les diagrammes état/transition, encore appelés machines à état, sont un outil visuel commode pour représenter des techniques d’interaction graphiques avec souris/clavier. Donner un automate état/transition, avec les événements et actions appropriées, pour les tâches et dispositifs suivants :

1) Construction d’un polygone fermé dans une zone de dessin.

2) Interaction avec un bouton. On pourra utiliser les évènements Enter et Leave qui correspondent à l'entrée et la sortie du curseur dans la zone du bouton. On rappelle que si l'on relâche le bouton de la souris alors que l'on est en-dehors de la zone du bouton, la sélection est annulée.

3) On complète le formalisme des machines à état en ajoutant un temporisateur. Lorsque l'on

appelle "*arm <ms>*" dans une action de la machine à états, l'évènement *Timeout* est émis à

l'expiration de ce délai (exprimé en millisecondes). L'action "*disarm*" permet de

désactiver l'émission d'un évènement ; elle n'a aucun effet si le temporisateur n'a pas été

armé. Une machine à états ne gère qu'un seul temporisateur : si "*arm <ms>*" est appelé

alors qu'il est déjà armé, le temporisateur est ré-armé avec le nouveau délai.

Construire la machine à états qui détecte des simple-clics et des double-clics. Un

double-clic est une séquence de deux clics séparés de moins de 200 ms. L'étendre aux

clics triples et quadruples.

4) Construire la machine à états qui décrit les flèches d'une barre de défilement : lorsque

l'on clique dessus, le document défile à vitesse fixe. Modifier cette machine pour que la

vitesse de défilement augmente progressivement, sauf lorsque l'on bouge la souris,

auquel cas la vitesse revient à sa valeur initiale.

**Problème.** Gestion de projets dans une entreprise. On vous demande de réaliser la conception d’une interface pour une application avancée de gestion de projets, en suivant la démarche proposée en cours. Les objectifs de cette application sont de permettre aux responsables de projets de suivre l’ensemble des projets qu’ils pilotent et de pouvoir réagir rapidement à des événements (faits) survenant dans l’évolution du projet, grâce aux fonctionnalités d’aide à la décision offertes par l’outil.

L’activité de gestion de projet est vaste, vous ne considèrerez que les tâches suivantes :

* Définition d’un projet que vous restreindrez aux attributs suivants : nom du projet, identifiant, responsable, participants, date de création, date prévisionnelle de fin, phases du projet, état, risques potentiels (sous forme d’un commentaire).
* Planification des phases d’un projet (diagramme de Gantt avec les différentes phases du projet présentées en ordonnée avec le déroulement temporel présenté en abscisse).
* Suivi de l’avancée des projets.
* Prise de décision s’appuyant sur le module d’aide à la décision de l’application :
	+ Obtention de statistiques sur un projet, un membre de projet.
	+ Recherche de causes relatives à un fait donné.
	+ Recherche de conséquences (effets) relatives à un fait donné.
	+ Proposition de correction suite à un risque identifié.

La notion de «fait» est rattachée à celle de base de connaissances utilisée par l’application. Elle est indépendante de l’interface. Seules les 4 propriétés suivantes d’un fait vous concernent : description du fait, projet cible, composant cible du projet, date d’apparition.

1) Suite à l’analyse des buts à atteindre et des postes de travail, que vous préciserez à votre convenance, dessinez le modèle de tâches SADT qui guidera votre conception.

2) Modélisez les éléments d’interface correspondant aux 2 tâches principales suivantes : définition d’un projet et prise de décision suite à l’arrivée des faits. Vous présenterez l’architecture des éléments d’interface à l’aide d’un ou plusieurs modèles PAC**.**