

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'Hamed BOUGUERRA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

---

Nature de l'examen : EMD  
Durée : 1h30  
Documents : Interdits  
Documents : Sujet

Module : ASD1  
Filière: LI3 & Tec Web3  
Année : 2010/2011  
Responsable : HADJIDJ D.

---

**Partie A (6 pt)**

1- Soient deux listes linéaires d'entiers, de têtes debut1 et debut2.  
Ecrire une fonction c++ permettant d'insérer la liste de tête debut2 dans la liste de tête debut1 après l'élément de valeur « val ».

**Partie B (14pt)**

Ecrire un programme c++ avec fonctions permettant d'effectuer les opérations suivantes :

- 1- Construire un tableau dynamique de M listes linéaires simples, composée chacune de N valeurs entières.
- 2- Afficher les valeurs de toutes les listes.
- 3- Inverser toutes les listes.
- 4- Afficher les valeurs de toutes les listes après l'inversion.

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'Hamed BOUGARA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

---

Nature de l'examen : EMD  
Durée : 1h30  
Documents : Interdits  
Documents : Sujet

Module : ASD1  
Filière: LI3 & Tec Web3  
Année : 2011/2012  
Responsable : HADJIDJ D.

---

**Exercice 1 (7 pt)**

Soit un tableau dynamique de M listes linéaires simples d'entiers.

Ecrire une fonction c++ et son appel pour supprimer tous les éléments de valeur égale à 'val' donnée.

**Exercice 2 (7 pt)**

Pour représenter deux polynômes, on considère deux listes linéaires de têtes P1 et P2, où chaque maillon contient un exposant et le coefficient correspondant.

En supposant que les listes sont triées dans l'ordre décroissant des puissances, écrire une fonction c++ et son appel pour construire la liste représentant la somme des deux polynômes.

**Exercice 3 (6 pt)**

Ecrire une fonction C++ et son appel pour partitionner une pile P de n entiers ( n pair) en deux piles P1 et P2 . La fonction doit être utilisable en même temps pour des piles représentées par tableaux et des piles représentées par listes linéaires.

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'Hamed BOUGUERRA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

Nature de l'examen : EMD

Module : ASD1

Durée : 1h30

Filière: LI3 & Tec Web3

Documents : Interdits

Année : 2011/2012

Documents : Sujet

Responsable : HADJIDJ D.

**Partie A (6 pt)**

Ecrire une fonction C++ permettant de partitionner une pile de  $n$  entiers ( $n$  pair) en deux piles P1 et P2 utilisable en même temps pour des piles représentées par tableaux et des piles représentées par listes linéaires.

Soient deux listes linéaires d'entiers triées, de têtes L1 et L2.

Ecrire une fonction c++ permettant de construire sans créer de nouveaux éléments une liste triée formée des éléments communs aux deux listes.

**Partie B (14pt)**

Soit un tableau dynamique de  $M$  listes linéaires simples d'entiers.

Ecrire une fonction c++ avec fonctions permettant d'effectuer les opérations suivantes :

5- Afficher les valeurs de toutes les listes.

6- Inverser toutes les listes.

7- Afficher les valeurs de toutes les listes après l'inversion.

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'Hamed BOUGARA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

Nature de l'examen : EMD

Module : ASD1

Durée : 1h30

Filière: LI3 & Tec Web3

Documents : Interdits

Année : 2012/2013

Documents : Sujet

Responsable : HADJIDJ D.

**Exercice 1 (10pt)**

Ecrire un programme qui effectue les opérations suivantes :

- construire une liste linéaire L1 contenant la description de processus (programmes) dans un système multi-utilisateurs. Chaque processus étant décrit par une structure St1 comportant un id (entier), un username (chaîne de caractères) et une priorité (entier de 1 à  $n$ ).
- construire à partir de L1 un tableau tab où chaque élément pointe sur une liste des processus de même priorité.

**Exercice 2 (5 pt)**

Soient deux piles d'entiers triées p1 et p2.

Ecrire une fonction permettant de fusionner ces deux piles. La fonction doit être utilisable en même temps pour des piles représentées par tableaux et des piles représentées par listes linéaires.

**Exercice 3 (5 pt)**

Ecrire une fonction C++ et son appel pour effectuer l'insertion en début d'une liste linéaire circulaire.

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'HAMED BOUGARA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

Nature de l'examen : ETLD

Module : ASD2 2012-2013

Responsable : D. HADJIDJ

**Exercice 1 (8 pts)**

Ecrire un programme c++ avec fonctions permettant d'effectuer les opérations suivantes.

- Créer deux arbres binaires de tri représentés chacun par un tableau statique.
- Fusionner sans répétition ces deux arbres dans une file représentée par un tableau dynamique.
- Afficher la file.

**Exercice 2 (6pts)**

Soit un arbre binaire représenté par un tableau statique.

Ecrire une seule fonction c++ permettant de retourner le nombre total des sommets, le nombre de sommets feuilles et le nombre de sommets internes et d'indiquer si l'arbre est de tri.

**Exercice 3(6 pts)**

On suppose avoir en mémoire centrale un graphe orienté représenté par une liste d'adjacence.

Ecrire une fonction permettant d'inverser les successeurs du sommet de degré maximal du graphe.

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'HAMED BOUGARA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

Nature de l'examen : ETLD

Module : ASD1 2013-2014

Responsable : D. Hadjidj &M. Sadou

**Exercice 1 ( 6pts)**

Soit A un tableau dynamique de M listes linéaires simples d'entiers.

Ecrire une fonction c++ permettant d'exploiter A pour retourner un tableau dynamique B constitué des M listes et indiquant le nombre d'éléments de chacune d'elles.

**Exercice 2 ( 7pts)**

Soit une liste linéaire L1 contenant la description de processus (programmes) dans un système multi-utilisateurs. Chaque processus étant décrit par une structure comportant un id (entier), un username (chaîne de caractères) et une priorité (entier de 1 à n).

Ecrire une fonction c++ permettant de supprimer de L1, tous les processus dont l'identificateur est 40.

**Exercice (7 pts)**

Soient deux piles d'entiers triées p1 et p2.

Ecrire une fonction c++ permettant de retourner une pile composée des éléments communs aux deux piles. La fonction doit être utilisable en même temps pour des piles représentées par tableaux et des piles représentées par listes linéaires.

NB : donner les types de structures utilisées et les appels à vos fonctions

Autres exercices

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'HAMED BOUGARA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

Module : ASD

Responsable : D. HADJIDJ

Exo1 (7 pt)

Ecrire un programme avec fonctions permettant de construire et d'afficher un tableau dynamique de N (N donné) listes linéaires d'entiers.

Exo2 (6pt)

Soit une liste linéaire de N entier de tête L1.  
Ecrire une fonction permettant d'inverser la première moitié de L1.

Exo3 (7pt)

Soient A et B deux ensembles triés d'entier représentés par des listes linéaires.  
Ecrire une fonction permettant de construire l'intersection de A et B dans un ensemble C en optimisant le temps de construction.

Partie A (5 pt)

Ecrire un programme C++ permettant de créer deux tableaux bidimensionnels entiers dynamiques A (N, N) et B (N, N), d'effectuer leur somme et de les afficher ainsi que le résultat.

Partie B (5 pt)

Soient trois listes linéaires d'entiers, de têtes D1, D2 et D3.  
Ecrire un programme C++ avec fonctions permettant de concaténer ces 3 listes.

Partie C (8 pt)

Ecrire une fonction C++ permettant de trier une liste linéaire d'entiers.

Partie D (4 pt)

Ecrire une fonction C++ permettant d'inverser une liste linéaire d'entier.

---

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'HAMED BOUGARA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

Module : ASD

Responsable : D. HADJIDJ

**Partie A (5 pt)**

1- Ecrire une fonction c++ permettant de permuter les valeurs minimales de deux tableaux bidimensionnels entiers A [N] [N] et B [N] [N] connaissant leurs indices lignes et colonnes.

**Partie B (15pt)**

Ecrire un programme c++ avec fonctions permettant d'effectuer les opérations suivantes :

- 1- Construire deux anneaux d'entiers.
- 2- Trier les éléments des deux anneaux
- 3- Fusionner sans répétition les éléments des deux anneaux dans une pile représentée par un tableau.
- 4- Afficher la piles et les anneaux.

**Partie A (7pt)**

Soit un tableau A [N] de listes linéaires d'entiers.

Ecrire une fonction c++ permettant de supprimer tous les éléments dont la valeur est supérieure à 50.

**Partie B (6 pt)**

Soient les déclarations suivantes :

Const int Max= ... ;

struct sommet

{int cle ;

int gauche, droit ;

};

Sommet groupesom[Max] ;

Ecrire une fonction permettant de construire un arbre binaire de tri représenté par le tableau groupesom.

**Partie C (7 pt)**

Soit un graphe orienté  $G = (V, E)$  représenté en mémoire centrale par une liste d'adjacence.

Ecrire une fonction permettant d'inverser dans cette structure les arcs ayant pour sommet d'origine le sommet S.

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITE M'HAMED BOUGARA –BOUMERDES  
FACULTE DES SCIENCES  
DEPARTEMENT D'INFORMATIQUE

Module : ASD

Responsable : D. HADJIDJ

**Exercice 1 (6 pts)**

2 - Ecrire une fonction C++ pour fusionner deux arbres de tri.

**Exercice 2(6 pts)**

Soit un arbre binaire représenté par un tableau.  
Ecrire une fonction permettant de vérifier si l'arbre est de tri.

**Exercice 3 (8pts)**

On suppose avoir en mémoire centrale un graphe orienté représenté par une liste d'adjacence.  
Ecrire une fonction permettant de supprimer tous les sommets de degré supérieur à 4.

**Partie A**

Ecrire un programme C++ avec fonctions pour effectuer les opérations suivantes :

- 1- Créer deux arbres de tri ayant des clés entières.
- 2- Afficher les deux arbres.
- 3- Créer deux anneaux linéaires à partir deux arbres de tri.
- 4- Afficher les deux anneaux.
- 5- Fusionner sans répétition les éléments des deux anneaux dans une pile représentée par un tableau.
- 6- Afficher la pile résultante.
- 7- Créer a partir de la pile résultante deux piles, l'une contenant les éléments paires l'autre les éléments impaires.
- 8- Afficher chacune des deux piles.

**Partie B**

1- On suppose avoir en mémoire centrale un graphe orienté représenté par une matrice d'incidence sommet-sommet.  
Ecrire une fonction permettant de construire à partir de cette matrice le même graphe en utilisant une liste d'adjacence.

2-En suppose avoir en mémoire centrale un graphe orienté représenté par une liste d'adjacence.  
Ecrire une fonction permettant de construire à partir de cette matrice le même graphe en utilisant une matrice d'incidence sommet-somme