

PARTIE 1 – INFORMATIQUE GÉNÉRALE

1. CULTURE « INTERNET »

Une ferme de liens (en anglais *link farm*) est une méthode utilisée pour augmenter artificiellement l'importance d'un site ou d'un groupe de sites dans les moteurs de recherches.

Le principe est de créer un certain nombre de sites se renvoyant les uns vers les autres. Les fermes de liens peuvent être créées à la main ou de manière automatisée. Cette technique, inventée pour tirer parti du page ranking de Google (et des moteurs de recherche fonctionnant sur le même principe) s'apparente à du spam. La société Google prévient d'ailleurs les webmestres qu'elle pénalise l'indexation des sites ayant recours à ce genre de système.



2. SYSTÈME DE NUMÉRATION

Ecrire les 20 premiers entiers positifs du système hexadécimal.

0 – 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – A – B – C – D – E – F – 10 – 11 – 12 – 13 – 14

Ecrire les 20 premiers entiers positifs du système octal.

0 – 1 – 2 – 3 – 4 – 5 – 6 – 7 – 10 – 11 – 12 – 13 – 14 – 15 – 16 – 17 – 20 – 21 – 22 – 23 – 24

3. CODAGE DES DONNÉES

	C	O	N	C	O	U	R	S		P	A	S	S	E	R	E	L	L	E
hexa	43	4F	4E	43	4F	55	52	53	20	50	41	53	53	45	52	45	4C	4C	45
décimal	67	79	78	67	79	85	82	83	32	80	65	83	83	69	82	69	76	76	69

4. CONVERSION

Calculer la valeur binaire de ces nombres en utilisant le complément à deux.

$(-52)_{10}$ à convertir en binaire : $(11001100)_2$.

$(-29)_{10}$ à convertir en binaire : $(11100011)_2$.

PARTIE 2 – ALGORITHMIQUE

```
Program liste_chainee ;
Type  lien = ^element ;
      element = record
        contenu : integer ;
        suivant : lien
      end ;
Var  debut : lien ;

Function recherche(debut : lien ; var prec,elt : lien ;
elt_cherche :integer) : boolean ;
(* elt pointe vers l'élément recherché, prec pointe vers
l'élément précédent. Ces deux pointeurs serviront dans la
procédure de suppression *)
Var  flag : boolean ;
Begin
  flag := false ;
  elt :=debut ;
  prec := nil ;
  While elt <> nil do
    Begin
      If elt^.contenu = elt_cherche then flag := true
    else
    begin
      prec := elt ;
    elt := elt^.suivant ;
      end ;
    End ;
    Recherche := flag ;
  End ;

Procedure insertion (var debut : lien ; elt_a_ajouter :
integer) ;
{*insertion en tête de liste*}
Var  nouveau : lien ;
Begin
  New(nouveau) ;
  nouveau^.contenu := elt_a_ajouter ;
  nouveaut^.suivant : debut ;
debut := nouveau
  end ;

  Procedure suppression (var debut : lien ;
elt_a_supprimer : integer) ;
```

```

Var prec,elt : lien ;
Begin
    If recherche(debut,prec,elt,elt_a_supprimer) then
begin
    If prec =nil then debut := elt^.suivant else
    prec^.suivant := elt^.suivant ;
Dispose(elt)
end
else
Writeln('elt n''existe pas')
End ;

```

PARTIE 3 – SYSTÈME D’INFORMATIONS

QUESTIONS DE COURS

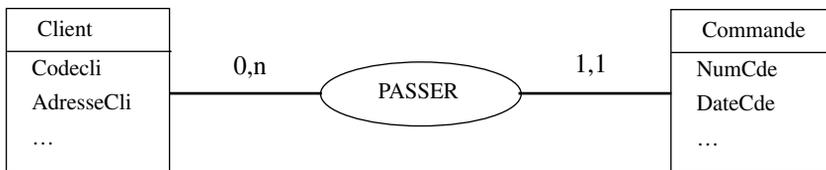
Une association hiérarchique met en relation deux entités dont l’une est père de l’autre. Les cardinalités de l’association sont 0,1 (ou 1,1) du côté fils et 0,n (ou 1,n) du côté père.

Une association non hiérarchique met en relation deux entités pour lesquelles il n’existe pas de lien de hiérarchie, les cardinalités de l’association sont 0,n (ou 1,n) pour chacune des entités.

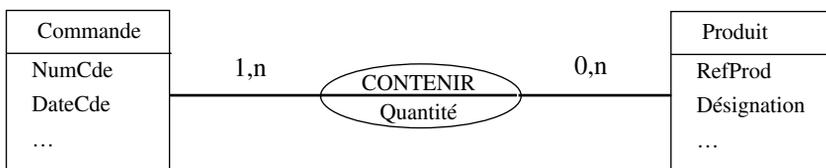
Lors de la transformation en MLD, les deux tables issues d’une association hiérarchique sont liées par une clé étrangère (la clé primaire de la table père devient clé étrangère dans la table fils).

Une association non hiérarchique est traduite par une nouvelle table dans le MLD. Cette table a pour clé primaire la concaténation des identifiants des deux entités que l’association liait dans le MCD.

Exemple Commande/Client



Exemple Commande/Produit



PROBLÈME

