

Cinématique de fichiers - COBOL
(Resp. A. VAILLY)

TP n° 4 : Gestion d'une vente aux enchères

Le dossier est à rendre pour le 2 avril 2010 au plus tard. Il sera déposé SUR LA PLATE-FORME MADOC uniquement, en format pdf exclusivement.

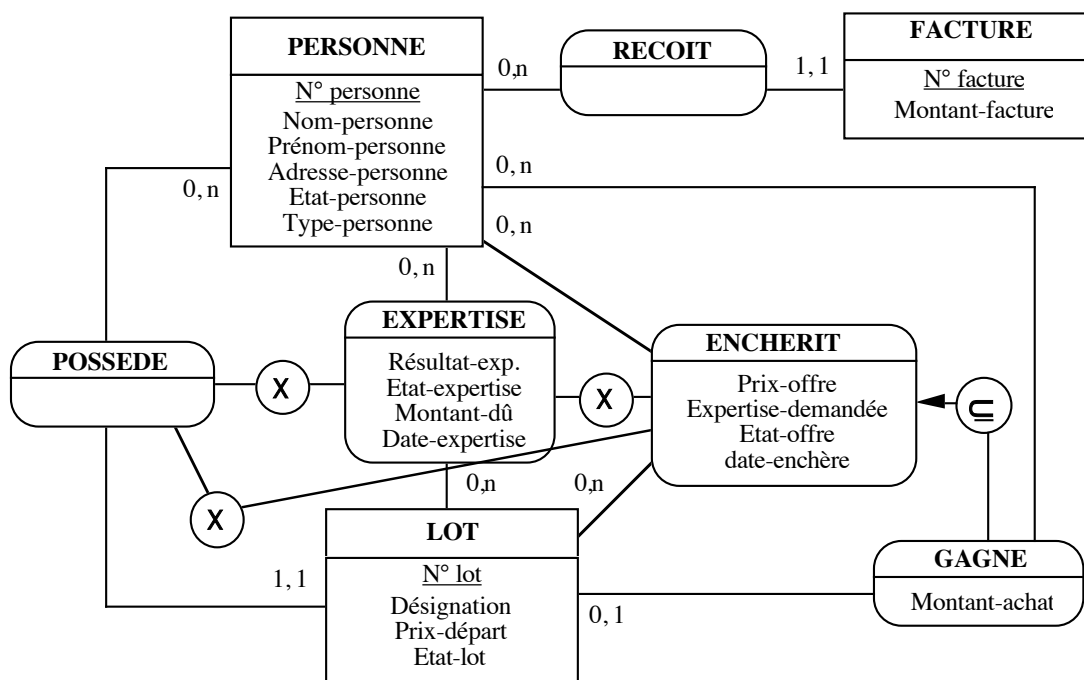
Les tests en salle machine auront lieu dans la semaine du 29 mars au 2 avril 2010. La présence des équipes au complet est requise.

Énoncé

Objectif : mise en place d'un mini logiciel de gestion de ventes aux enchères au sein d'un magasin de timbres, logiciel constitué d'un programme principal et de sous-programmes.

1- Description de la structure de données :

On se situe dans le cadre du développement d'une application de gestion de ventes aux enchères par correspondance d'un magasin de timbres. Chaque timbre, chaque client, chaque facture... est décrit. Le système d'information de cette application peut être décrit par le modèle entités associations propriétés suivant :



Les personnes peuvent être (ce sont les valeurs autorisées pour la propriété *État-personne*) « nouveau », « parti », « ancien », « mauvais payeur » ou « bon client ». Il peut s'agir (ce sont les valeurs

prises par la propriété *Type-personne*) d'un « vendeur », d'un « expert » ou d'un « acheteur ». Les lots peuvent prendre les états suivants : « proposé », « en vente », « retiré », « gagné » ou « vendu » (la facture a été réglée). Les expertises sont soit « anciennes » (le certificat d'expertise est fourni par le vendeur, avant la vente ; le montant de l'expertise est alors égal à 0), soit « en cours » (une demande d'expertise a été faite ; le résultat n'a pas encore été fourni), soit « nouvelle » (la demande émane du client qui a remporté l'enchère ; l'expertise a été demandée par le magasin PHILDEX ; le résultat de l'expertise a été fourni ; il y a un montant d'expertise non nul). Les enchères sont d'abord (durant toute la vente aux enchères) « en cours ». Elles seront ensuite (à la fin de la vente) soit « gagnante » (le prix proposé est le plus élevé), soit « perdante » (il y a un meilleur enchérisseur).

Seuls les experts (les personnes de type expert) peuvent délivrer une expertise. Seuls les vendeurs ou les acheteurs peuvent enchérir. Les vendeurs ne peuvent proposer d'enchère sur leurs propres lots. Par contre, ils peuvent le faire sur d'autres lots. Les lots sont la propriété des vendeurs. Une personne qui possède un lot est donc obligatoirement un vendeur.

Cette structure de données peut être matérialisée par six fichiers :

- fichier des personnes (*fpers*), **séquentiel indexé**, de clé primaire le numéro de personne (NOPERS), ayant la structure suivante :

<i>NOPERS</i>	<i>NOMPERS</i>	<i>PREPERS</i>	<i>ADPERS</i>	<i>ETATP</i>	<i>TYPEP</i>
entier 4 chiffres	20 alpha. num.	15 alpha. num.	25 alpha. num.	1 lettre	1 lettre
9 (4)	X (20)	X (15)	X (25)	A	A

La variable *ETATP* peut prendre les valeurs A (ancien), B (bon client), M (mauvais), N (nouveau) ou P (parti). La variable *TYPEP* peut prendre les valeurs A (acheteur), E (expert) ou V (vendeur).

- fichier des factures (*ffact*), **séquentiel indexé**, de clé primaire le couple (numéro de facture, numéro d'acheteur) (NUMFAC, NUMACH)) et de clé secondaire le numéro d'acheteur (NUMACH), ayant la structure suivante :

<i>NUMFAC</i>	<i>NUMACH</i>	<i>MONTFAC</i>
entier 6 chiffres	entier 4 chiffres	réel 8 chiffres, dont 2 décimales
9 (6)	9 (4)	9 (6) V 99

- fichier des lots (*flots*), **séquentiel indexé**, de clé primaire le numéro de lot (NUMLOT) et de clé secondaire le numéro du propriétaire (NUMPROP), de structure :

<i>NUMLOT</i>	<i>DESIGN</i>	<i>PRIXDEP</i>	<i>ETATL</i>	<i>NUMPROP</i>
entier 4 chiffres	30 alpha. num.	réel 6 chiffres, dont 2 décimales	1 lettre	entier 4 chiffres
9 (4)	X (30)	9 (4) V 99	A	9 (4)

la variable *ETATL* pouvant prendre les valeurs E (en vente), G (gagné), P (proposé), R (retiré) ou V (vendu).

- fichier des expertises (*fexpert*), **séquentiel indexé**, de clé primaire le couple (numéro de lot, numéro d'expert) (NUMLOT, NUMEXP), de structure :

<i>NUMLOT</i>	<i>NUMEXP</i>	<i>RESULT</i>	<i>ETATE</i>	<i>MONTANT</i>	<i>DATEXP</i>
entier 4 chiffres	entier 4 chiffres	1 lettre	1 lettre	entier 4 chiffres	8 alpha. num.
9 (4)	9 (4)	A	A	9 (4)	X (8)

la variable ETATE pouvant prendre les valeurs A (ancienne), E (en cours) ou N (nouvelle). Le résultat de l'expertise est décrit sous forme d'une lettre (A pour authentique, F pour faux). Lorsque le résultat de l'expertise n'est pas encore connu, la variable RESULTAT n'a pas de valeur.

- fichier résultat (*fres*), **séquentiel indexé**, ayant pour clé primaire le couple (numéro de lot, numéro d'acheteur) (NUMLOT, NUMACH), avec une clé secondaire, NUMACH. La structure de ce fichier est la suivante :

<i>NUMLOT</i>	<i>NUMACH</i>	<i>MONTANT</i>
entier 4 chiffres	entier 4 chiffres	réel 6 chiffres, dont 2 décimales
9 (4)	9 (4)	9 (4) V 99

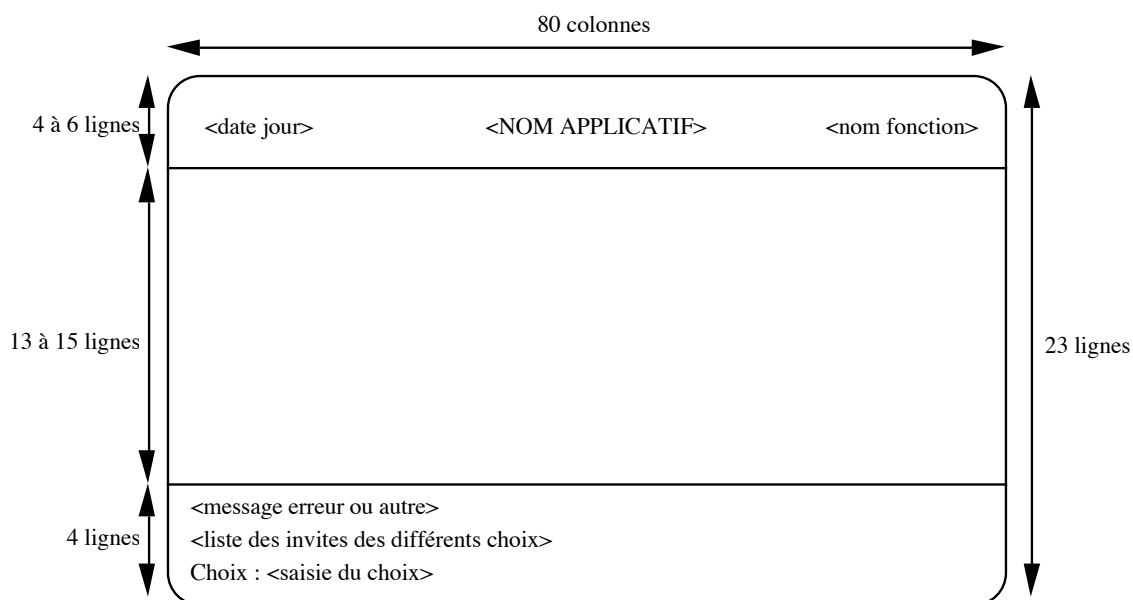
- fichier des enchères (*fench*), **séquentiel indexé**, de clé primaire le couple (numéro de lot, numéro d'acheteur) (NUMLOT, NUMACH), ayant deux clés secondaires, le numéro de lot (NUMLOT) pour l'une, le numéro de l'acheteur (NUMACH) pour l'autre. Il a pour structure :

<i>NUMLOT</i>	<i>NUMACH</i>	<i>AEXP</i>	<i>ETATO</i>	<i>MONTANT</i>	<i>DATENCH</i>
entier 4 chiffres	entier 4 chiffres	1 lettre	1 lettre	entier 4 chiffres	8 alpha. num.
9 (4)	9 (4)	A	A	9 (4)	X (8)

la variable ETATO prenant la valeur G (gagnante), H (gagnante par tirage au sort), P (perdante) ou Q (perdante par tirage au sort). La variable AEXP est une variable logique. Elle vaut O (une expertise est demandée) ou N (une expertise n'est pas demandée).

2) Interface du logiciel :

Tous les écrans seront standardisés. Ils seront tous composés de trois parties, selon le modèle ci-joint :



Les différentes parties jouent des rôles différents :

- entête (partie supérieure) : 4 à 6 lignes
désignation de l'application, fonction en cours, date du jour et ligne de séparation

- corps (partie médiane) : 13 à 15 lignes
affichage de résultats, saisies autres que les invites...
- bas de page (partie inférieure) : 4 lignes
ligne 19 : ligne de séparation
ligne 20 : ligne pour les messages provenant d'erreurs de saisie, de manipulation...
ligne 21 : ligne d'invite indiquant les différents choix de poursuite
ligne 22 : ligne pour saisir le caractère représentant le choix.

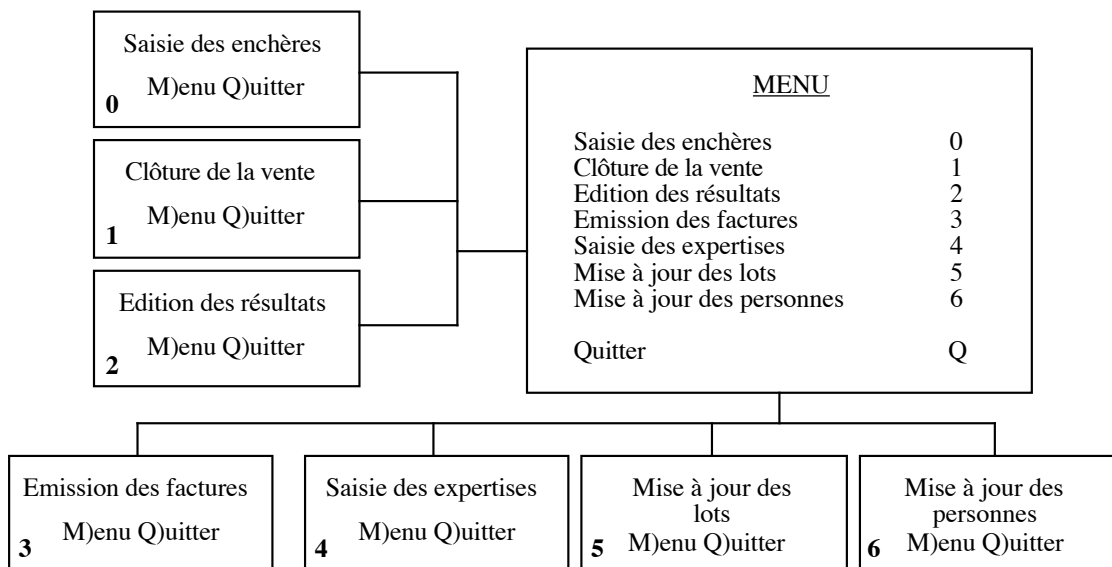
L'écran de présentation du menu sera structuré différemment des autres : présentation des différents choix et saisie du choix dans la partie médiane, partie inférieure ne contenant que le message d'erreur de saisie.

3) Structuration du logiciel :

L'application comprend les fonctionnalités suivantes :

- F0 : saisie des enchères,
- F1 : clôture de la vente,
- F2 : édition des résultats de la vente (avec liste des lots vendus, liste des lots invendus et liste des lots vendus à expertiser),
- F3 : émission des factures et envoi des lots aux clients connus,
- F4 : saisie des expertises (avec édition des factures et envoi éventuel des lots),
- F5 : mise à jour des lots (ajout, modification et/ou suppression), avec saisie éventuelle des expertises correspondantes,
- F6 : mise à jour des personnes (ajout, modification et/ou suppression) SANS saisie des lots mis en vente.

L'architecture fonctionnelle du logiciel à développer est une arborescence :



Chaque fin d'exécution d'une feuille de cet arbre doit permettre à l'utilisateur de choisir soit de revenir au menu, soit de quitter l'application.

4) Saisies :

Les données saisies sont supposées pleines d'erreurs. Toutes les données saisies devront donc être vérifiées. Les données erronées donneront lieu à affichage de messages d'erreurs appropriés (dans la partie inférieure de l'écran), effacement des données erronées (afin de permettre une resaisie) et, après la

nouvelle saisie, effacement du message d'erreur précédent. Le nombre d'essais sera limité à trois pour chaque saisie.

Le choix de la partie du logiciel à exécuter (menu) fourni par l'utilisateur sera vérifié AVANT de lancer l'exécution.

5) Travail à faire :

5.1) organisation : le travail est exécuté par des équipes de quatre (exceptionnellement cinq). Chaque équipe désignera, dès le début, un chef de projet.

5.2) travail de l'équipe : tous les membres de l'équipe devront analyser et programmer le logiciel. Une organisation avec un programme principal (pour l'affichage du menu) et des sous-programmes externes, un par module, sera mise en place. Le programme principal devra être programmé en premier. Il fera "normalement" appel aux divers sous-programmes externes, un par fonction, selon le principe développé en TD (cf. TD n° 4, programme P4). Même si une fonction n'est pas encore prête, le sous-programme externe correspondant devra donc être présent dès le début. Il comportera AU DEBUT un écran "par défaut" qui affichera, en partie médiane, un message indiquant que la fonction n'est pas encore disponible. Le "vrai" sous-programme une fois testé, ce sous-programme par défaut disparaîtra au profit du "vrai".

Chaque équipe devra analyser et programmer les fonctions ci-après :

F0 : à chaque fois que la maison PHILDEX reçoit une enchère, elle procède à son enregistrement. Si l'acheteur est inconnu, il y a d'abord enregistrement de celui-ci (il est décrété nouveau), puis saisie de l'enchère. Celle-ci est immédiatement comparée aux enchères concurrentes et, en particulier, à celle(s) qui était(en)t gagnante(s). Si la nouvelle offre est perdante, elle est mémorisée en tant que telle (sa variable ETATO prend la valeur P). Si elle est gagnante, elle est mémorisée avec une variable ETATO à G. Elle sera gagnante si le prix d'achat proposé est supérieur au montant de l'offre gagnante précédente ou s'il est identique. Il peut donc y avoir des ex-aequos. Ceux-ci seront tranchés lors de la clôture de la vente. La (ou les) gagnante(s) précédente(s) sera (seront) déclarée(s) perdante(s).

Une enchère concernant un lot inexistant sera refusée. Une enchère passée par le propriétaire du lot sera également refusée. Une enchère en double (même personne, même lot) ne sera acceptée que si elle représente une offre supérieure à la précédente. Son montant doit donc être supérieur à celui de l'offre identique déjà mémorisée. La seconde remplacera alors la plus ancienne. Un acheteur parti ne peut plus faire d'offres.

F1 : lorsque la vente est terminée, les enchères gagnantes sont recensées. Les ex-aequos sont mis en évidence et tirés au sort (les modalités pratiques de tirage au sort sont laissées à la libre appréciation des concepteurs de l'application). Les ex-aequos perdants sont mémorisés dans le fichier des enchères avec la valeur Q pour ETATO. L'ex-aequo gagnant est mémorisé dans le même fichier des enchères avec la valeur H.

Le prix de vente d'un lot est établi comme suit :

- si l'enchère est unique, le prix de vente est égal au prix de départ (PRIXDEP) du lot ;
- s'il y a eu plusieurs enchères, le prix de vente est égal au montant de l'enchère arrivée seconde, majoré de 10 %. Ainsi, avec un lot ayant « produit » deux enchères, une de 100 et une seconde de 150, le prix de vente du lot sera de 100 (le prix de l'offre perdante) plus 10 (les 10 % d'augmentation). Le lot sera acheté à 110 par la personne qui avait proposé 150.

Le résultat de la vente est stocké dans le fichier des résultats *fres*. Si le gagnant avait demandé une expertise, celle-ci est lancée. Il y a alors création d'une nouvelle ligne dans le fichier des expertises *fexpert*, la variable ETATE prenant la valeur E (expertise en cours). Les autres variables n'ont pas de valeur.

La variable ETATL est modifiée en conséquence : les lots ayant été l'objet d'une enchère sont considérés comme gagnés (la variable prend la valeur G), les lots sans enchère sont considérés comme retirés (la variable prend la valeur R).

F2 : trois listes sont produites, une des lots gagnants à expertiser, une des lots remportés et une des lots invendus. La liste des lots à envoyer aux experts comprend deux parties, une première qui correspond à la liste, en ordre croissant, des lots à envoyer et une seconde qui correspond à une liste PAR EXPERT, triée en ordre croissant sur le numéro d'expert en majeur, puis en mineur sur les numéros de lots à envoyer.

La liste des lots invendus, elle, comprend deux parties, une première qui correspond à la liste, en ordre croissant, des lots à renvoyer aux vendeurs et une seconde qui correspond à une liste PAR

VENDEUR, triée en ordre croissant sur le numéro de vendeur en majeur, puis en mineur sur les numéros de lots à retourner. Les lots invendus sont les lots pour lesquels PHILDEX n'a reçu aucune enchère.

La dernière liste, celles des lots remportés, est analogue à la précédente. Elle correspond, en fait, à la différence entre les lots à vendre et les lots invendus. Sa constitution, en deux parties, est similaire à la précédente. Les lots figurant dans la première liste doivent figurer dans cette dernière.

F3 : cette fonction permet de préparer l'expédition des lots achetés par les clients. Seuls les lots des clients qui n'ont pas demandé d'expertise sont traités ici. Si un client a obtenu trois lots et s'il a demandé une expertise pour un seul, ses trois lots seront mis en attente.

Il y a d'abord calcul du montant total des achats à partir du prix de vente de chaque lot remporté. Ce montant total est conservé dans un fichier temporaire à définir, ouvert en création. Il y a ensuite affichage, client par client, des numéros de lots remportés par lui, puis saisie des frais de port. Ces frais sont ajoutés au montant total des achats, ce qui donne le montant total facturé. Celui-ci est enregistré dans le fichier *ffact*, avec le numéro de l'acheteur. Le numéro de facture est calculé automatiquement, relativement à l'acheteur (il y a la facture 1, la facture 2... du client XXX ; la facture 1 du client YYY...).

L'écran suivant servira de support au dialogue :

EXPEDITION DES LOTS

N° acheteur

Lots remportés

Frais de port

Montant total facture

F4 : lorsqu'une expertise demandée par l'acheteur arrive au magasin (elle est envoyée par l'expert avec le résultat –authentique ou faux- et le montant à payer), elle est saisie. Il s'agit en fait d'une modification des champs RESULT, ETATE –cette variable prend la valeur N-, MONTANT et DATEXP.

Si l'expertise qui arrive n'existe pas dans le fichier *fexpert*, elle n'est pas saisie. Si le lot examiné est faux, la vente est annulée. Le lot est considéré comme retiré (ETATL est mis à R).

Le client qui avait demandé cette expertise recevra ses lots (y compris ceux en attente) si TOUTES les expertises en cours sont arrivées. Sinon, il devra encore attendre. Quand tout est là, le prix total est calculé, avec deux composantes, le prix de vente de tous les lots (on procède comme dans la fonction F3) et le prix des expertises. La somme des deux constitue le montant des achats. Il y a ensuite affichage des lots achetés, saisie des frais de port... comme dans F3.

La différence, à ce niveau, réside dans le nombre de clients traités. Dans F3, on passe en revue tous les clients. Dans F4, on ne s'occupe que des clients qui ont demandé l'expertise qui est saisie.

F5 : le fichier des lots peut être mis à jour (qu'il s'agisse d'un ajout, d'une suppression ou d'une modification). Un nouveau lot sera ajouté (la variable ETATL prend la valeur P) avec, le cas échéant, un ajout de l'acheteur s'il n'existe pas encore. Le lien de propriété sera dans tous les cas (acheteur ancien ou acheteur nouveau) mis à jour. Le numéro des lots sera fourni par l'utilisateur.

La suppression est une suppression logique, le lot étant marqué retiré (ETATL = R).

Toutes les informations (sauf le numéro de lot) sont modifiables, y compris le lien de propriété. Dans ce dernier cas, s'il y a eu erreur de propriétaire, le numéro du nouveau devra remplacer l'ancien. ATTENTION TOUTEFOIS À LA MISE À JOUR DU CHAÎNAGE CORRESPONDANT À LA CLÉ SECONDAIRE.

F6 : le fichier des personnes peut être mis à jour (qu'il s'agisse d'ajout, d'une suppression ou d'une modification). Qu'il s'agisse d'un expert, d'un vendeur ou d'un acheteur, il est possible d'en ajouter un sans définir le ou les lots en rapport. Un vendeur pourra donc être enregistré sans que ses lots le soient. Les nouveaux (acheteurs, experts, vendeurs) sont codifiés en tant que tels. La variable ETATP, dans leur cas, prendra la valeur N. Les numéros de personne sont fournis par l'utilisateur et non calculés par le système.

La suppression est une suppression logique. La variable ETATP de celui ou de celle qui s'en va est mis à la valeur P (pour parti).

Toutes les informations (sauf celle du numéro) sont modifiables. Cette fonction permet également de rectifier des erreurs d'affectation d'une expertise. On procède alors comme ceci : dans un premier temps, l'expertise faussement attribuée à la personne est ré-affectée au bon expert (il s'agit alors de créer cette nouvelle expertise). Dans un second temps, la mauvaise affectation est supprimée. Prenons un exemple pour illustrer ce traitement. Nous avons dans la base une expertise du lot 33 effectuée par Mr CCC. C'est une erreur. L'expertise a en fait été réalisée par Mme GGG. Nous allons donc, dans un premier temps, créer l'expertise correcte (33, GGG) puis, dans un second temps, supprimer l'expertise fausse (33, CCC).

5.3) rôle du chef de projet : le chef de projet est chargé, plus spécifiquement, de l'intégration des sous-programmes "dans" le logiciel. C'est lui qui crée le programme principal, teste son fonctionnement, remplace les sous-programmes "par défaut" par les "vrais", dès que possible. Le chef de projet doit aussi piloter l'équipe, en surveillant l'avancement du travail, en aidant, éventuellement, un co-équipier.

C'est, enfin, le chef de projet qui crée un répertoire GROSTP (avec les droits nécessaires pour que chaque membre de l'équipe puisse y accéder), y place les jeux d'essais, y remet éventuellement ces jeux d'essais s'ils sont altérés -en allant les chercher sur l'Intranet). L'intégration des sous-programmes dans ce répertoire sera effectuée sous la responsabilité du chef de projet. Les tests de non-régression (ie. globaux) seront réalisés par lui.

5.4) répartition des tâches au sein de l'équipe : soit une équipe quelconque, composée de cinq étudiants, appelés A, B, C, D et E. A est le chef d'équipe. La répartition des tâches peut être la suivante :

A : (chef de projet) création du programme principal, intégration, tests de non-régression, analyse et programmation F2

B : analyse et programmation F1, tests unitaires, import dans GROSTP

C : analyse et programmation F0, tests unitaires, import dans GROSTP

D : analyse et programmation F3 et F4, tests unitaires, import dans GROSTP

E : analyse et programmation F5 et F6, tests unitaires, import dans GROSTP

D'autres organisations sont possibles. Le "KIFEKOUA" sera défini dès le début du TP et fourni aux enseignants.

6) "documents" à rendre :

Chaque équipe rendra un seul dossier, en format pdf. Il devra comporter une partie analyse algorithmique (une douzaine de pages au maximum) et une partie jeux d'essais (avec une description des essais prévus par l'équipe, avec les résultats escomptés, rédigée sur un maximum de quatre pages).

Le listing de chaque programme (programme principal, sous-programmes externes), le "chemin" Linux indiquant où sont situées les sources, ainsi que la ligne de compilation seront fournis en annexe du dossier.

Ce dossier sera déposé sur la plate-forme MADOC, dans le répertoire créé à cet effet. La personne qui déposera ce dossier sera le chef de projet, le dépôt se faisant sous son SEUL nom.