Travaux Pratiques d'Unix n°10

Cours d'Informatique de Deuxième Année

—DEUG Sciences S4-MI—

Un système de Facturation

L'objectif de ce TP est de créer un script en Bourne shell permettant de simuler un système simplifié de facturation. À la fin des trois heures de TP, ce script sera envoyé par mail au responsable du TP.

Ce système est constitué de deux fichiers. Le premier nommé `encours` contient les factures en cours de paiement. Le format du fichier `encours` est le suivant :

```
num_facture nom_client date_emission liste_produits montant_facture
```

Chaque champ est séparé par une tabulation. La liste des produits est constituée d'une liste de couples `nom_produit prix` séparés par une virgule et où `nom_produit` est séparé de `prix` par un espace.

Un second fichier nommé `archive` contient un archivage des factures qui ont été payées. Le format de ce fichier est le suivant :

```
num_facture nom_client date_emission liste_produits montant_facture date_reglement
```

Chacune des lignes de ce fichier est identique à celle du fichier `encours` à laquelle on rajoute la date de règlement. Deux fichiers d'exemples vous seront donnés par les responsables du TP.

▸ Exercice 1.

1. *Réaliser un menu permettant de choisir entre plusieurs commandes. Pour cela vous utiliserez la commande* `case` *de format suivant :*

   ```
   case <variable> in
      <motif1>) <liste de commandes> ;;
      <motif2>) <liste de commandes> ;;
      ... ;;
   esac
   ```

2. *Écrire une commande qui affiche la date courante au format* `JJ/MM/AAAA`.

3. *Créer une commande permettant de rajouter une facture en cours de règlement. La procédure de création impose de pouvoir saisir indépendamment les différents champs (dont plusieurs produits possibles), d'insérer automatiquement la date au format précédent et de calculer automatiquement le montant de la facture.*

4. *Écrire une commande qui affiche la liste des numéros des factures déjà payées.*

5. *Écrire une commande qui affiche la liste des noms (sans redondance) des clients ayant des factures impayées.*

6. *Écrire une commande qui affiche la somme des montants de toutes les facturations en cours.*

7. *Écrire une commande qui affiche toutes les factures en cours émises avant une date qui sera entrée au clavier.*

8. *Créer une commande permettant d'archiver une facture au moment de son paiement. La facture sera sélectionnée à partir de son numéro de facture entré au clavier.*