U.F.R. de Mathématiques et Informatique **Année 1995-1996**
Département d'Informatique **Juin**

Module de TELEINFORMATIQUE
Durée 2h

Tous documents manuscrits autorisés.
Notes dactylographiées distribuées en Cours et en TD autorisées.

Exercice 1 (4 points)

Donnez la règle de réécriture à insérer dans le fichier sendmail.cf dans les cas suivants:

1 - supprimer une suite de plus de deux points consécutifs.

Exemple : l'adresse

Jean**..**Durand@u-bordeaux...fr

sera transformée en

Jean**.**Durand@u-bordeaux.fr

2 - supprimer une suite de plus de deux points précédant le @.

Exemple : l'adresse

Jean**...**Durand@u-bordeaux....fr

sera transformée en

Jean**.**Durand@u-bordeaux....fr

3 - supprimer la première suite de plus de deux points.

Exemple : l'adresse

Jean**...**Durand..@u-bordeaux...fr

sera transformée en

Jean**.**Durand..@u-bordeaux...fr

Exercice 2 (4 points)

Vous arrivez dans une entreprise de type PME. Cette entreprise n'a aucune connaissance d'INTERNET autre que celle transmise par les médias.

1 - Décrivez l'information que vous ferez aux dirigeants. Eventuellement, donnez les points de ressources WWW que vous utiliserez. La réponse ne doit pas dépasser 1/4 de page.

2 - Décrivez comment vous organiserez son raccordement. On ne demande pas ici comment s'effectuera l'installation des raccordements ni les fichiers. La réponse ne doit pas dépasser 1/4 de page.

Exercice 3 (7 points)

Un serveur est installé sur le port 1254. L'application nécessite que chaque client soit redirigé sur un port fixé par le serveur.

1 - Quelle différence y-a-t-il entre les implémentations UDP et TCP ?

Dans la suite on supposera que l'application fonctionne en UDP.

2 - Décrire le protocole que vous choisissez pour le dialogue client/serveur.

On suppose que l'application est donnée par le code C fourni en annexe.

3 - Quel est le code à insérer dans le client pour réaliser cette opération?

4 - Quel est le code à insérer dans le serveur pour réaliser cette opération?

Exercice 4 (5 points)

Les trames RR et REJ de HDLC permettent toutes deux d'accuser réception de trames correctement reçues jusqu'au numéro N-1.

1 - Quelle est la différence entre ces deux trames de contrôle?

2 - Soit ε un nombre très petit. On suppose que le temps d'échange entre deux stations A et B est régi par la règle suivante : le temps de transmission de A vers B est $1-\varepsilon$ celui de B vers A est $1+\varepsilon$. On suppose également que la liaison a déjà été établie. Les trames échangées sont les suivantes :

A envoie les informations A_1 au temps 1, A_2 au temps 2 , A_3 au temps 4

B envoie les informations B_1 au temps 1, B_2 au temps 3

Décrire les trames HDLC échangées entre les stations A et B dans les scénarios suivants.

S1 - Il n'y a pas de perte ni d'erreur dans les trames.

S2 - La trame A2 est perdue.

S3- La trame A1 est erronée.

ANNEXE (2pages)

Code client/serveur donné en TD

/* client_udp.c */

```
/* A lier pour creer un processus de lancement d'un client udp */
/* Appel de la commande avec les parametres : machine port ... */

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/types.h>
#include<netdb.h>
#include<sys/socket.h>
#include<netinet/in.h>

int creer_socket(int, int *, struct sockaddr_in *);
struct sockaddr_in adresse_serveur, adresse_client;

main(int argc, char *argv[])
{
 int port, socket_client;
 struct hostent *hp;

 if (argc<3)
 {
  fprintf(stderr,"erreur de parametre\n");
  exit(2);
 }
 /* recherche de l'adresse Internet de la machine du serveur */
 if ((hp=gethostbyname(argv[1])) == NULL)  {
  fprintf(stderr,"machine %s inconnue\n",argv[1]);
  exit(2);
 }
 /* creation/attachement de la socket du client sur un port quelconque */
 port=0;
 if ((socket_client=creer_socket(SOCK_DGRAM,&port,&adresse_client))==-1)
 {
  fprintf(stderr,"creation de socket du client impossible\n");
  exit(2);
 }
 printf("Client sur le port %d\n", ntohs(adresse_client.sin_port));
 /* preparation de l'adresse du serveur */
 adresse_serveur.sin_family = AF_INET;
 adresse_serveur.sin_port = htons(atoi(argv[2]));
 memcpy(&adresse_serveur.sin_addr.s_addr, hp->h_addr, hp->h_length);
 /* appel de la fonction specifique au service */
 client_service(socket_client, &adresse_serveur, argc-3, argv+3);
}
```

/* serveur_udp.c */

```
/* Lancement d'un serveur udp : serveur_udp port commande */

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<errno.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in.h>

int creer_socket(int, int *, struct sockaddr_in *);

main(int argc, char *argv[])
{
 struct sockaddr_in adresse;
 int port, desc_socket;

 if (argc!=3)
 {
  fprintf(stderr,"erreur de parametre\n");
  exit(2);
 }
 /* test d'existence du programme a executer pour le service */
 if (access(argv[2],X_OK)!=0)
 {
  fprintf(stderr,"fichier %s non executable\n",argv[2]);
  exit(2);
 }
 /* detachement du terminal */
 if (fork()!=0) exit(0);
 setsid();
 printf("serveur de pid %d lance\n",getpid());
 port = atoi(argv[1]);
 /* creation socket d'ecoute */
 if ((desc_socket=creer_socket(SOCK_DGRAM,&port,&adresse))==-1)
 {
  fprintf(stderr,"creation de socket impossible\n");
  exit(2);
 }
 dup2(desc_socket,STDIN_FILENO);
 close(desc_socket);
 execlp(argv[2],argv[2],NULL);
}
```