Exercice -2- (8 points)

- Écrire vos programmes en utilisant le langage C ou le langage C++.
- Respecter les noms attribués au programme (*exo2tp3.c*), fichier d'entrée *(in.txt)* et fichier de sortie *(out.txt)* ; ainsi que le format d'affichage en sortie.
- Compiler sur une des machines du laboratoire (1340) et assurer vous que vous utilisez la bonne version du compilateur gcc : La commande: « gcc –v » doit produire le résultat suivant :

  Reading specs from /usr/lib/gcc-lib/i386-redhat-linux/egcs-2.91.66/specs
  gcc version egcs-2.91.66 19990314/Linux (egcs-1.1.2 release).

- Prendre en compte que si vos programmes ne compilent pas ou ne font pas toutes les choses demandées dans la spécification, la note sera de 0/8.
- Inclure des commentaires dans vos programmes et bien les (programmes) indenter.
- Remettre un document (3 pages maximum) expliquant la conception de vos programmes (ne faites pas un copié collé de vos commentaires).

On vous demande d'écrire un programme qui va vérifier que les liens (URL) dans une page WEB existent.

En gros, pour vérifier l'existence d'une page WEB, on se branche avec des sockets INTERNET (stream) au PORT #80 du serveur où se trouve la page et on envoie une ligne de commande suivie d'une ligne blanche:

HEAD <chemin de la page> HTTP/1.0

Une SOCKET s'utilise comme un fichier normal UNIX avec un "fd" sur lequel on peut faire des lectures (read) et aussi des écritures (write). Le serveur nous retourne sur la même socket une entête. Ensuite, le serveur ferme la connexion. Tout l'échange se fait avec des chaînes de caractères lisibles. Vous pouvez expérimenter avec TELNET. Voici un exemple réel où on va chercher la page de base "/" du site diro:

---

dift3820@ phobos> ~/>telnet **www.iro.umontreal.ca 80**
Trying 132.204.24.95...
Connected to www.iro.umontreal.ca.
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Thu, 11 Jul 2002 13:47:49 GMT
Server: Apache/1.3.14 (Unix) mod_ssl/2.7.1 OpenSSL/0.9.6
Last-Modified: Wed, 03 Jul 2002 14:52:11 GMT
ETag: "2006-10fd-3d230f9b"
Accept-Ranges: bytes
Content-Length: 4349
Connection: close
Content-Type: text/html

Connection closed by foreign host.

N^0 du port

Nom du serveur

Les deux informations recherchées dans cet exercice.

Fichiers fournis

- **in.txt**: un ensemble de liens (ce fichier peut contenir de 1 à 999...999 liens). Chaque ligne représente une requête. La première colonne contient le nom du serveur sur lequel vous allez établir la connexion telnet sur le port 80, la seconde colonne contient la page web à extraire. Ces deux colonnes sont séparées par une tabulation et vous pouvez supposer que la taille de chaque colonne ne dépassera pas 256 caractères.

- **out.txt**: donne le format d'affichage en sortie ainsi que les résultats des tests des liens se trouvant dans le fichier « in.txt ».Chaque ligne représente le résultat de la requête. Ce résultat doit être extrait de la première ligne se trouvant dans la réponse formulée par le serveur (voir le précédent exemple).