



**POLYTECHNIQUE
MONTRÉAL**

LE GÉNIE
EN PREMIÈRE CLASSE

INF3405 – Réseaux informatiques
TP 3

Frédéric Quenneville, 1714871
Samuel Rondeau, 1723869

Présenté à
Saida MAAROUFI

17 mai 2016
Polytechnique de Montréal

Poste de travail

L4708-09

Question 1

15

```
C:\Windows\system32\cmd.exe

getaddrinfo a reussi!
Adresse # 1
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::30cd:f7ba:10be:6928
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 2
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::284c:b5e0:c397:d8f7
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 3
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::e85f:f0e3:4016:723e
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 4
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::a929:940b:e737:b708
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 5
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::3d37:c3d8:d53c:a0c6
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 6
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::a5d2:431:6af5:71f
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 7
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::1900:623:928:4405
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 8
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 132.207.29.109
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 9
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 169.254.216.247
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 10
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 192.168.68.1
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 11
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 192.168.111.1
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 12
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 192.168.24.1
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 13
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 192.168.112.1
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 14
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 192.168.64.1
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 15
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: 2002:84cf:1d6d::84cf:1d6d
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Appuyer une touche pour finir...
```

Question 2

On peut retrouver plusieurs adresses IP différentes pour une même machine si elle possède plusieurs adaptateurs réseau et/ou cette dernière fait partie de plusieurs sous-réseaux.

Question 3

Classe	Adresse min	Adresse max	Masque
A	1.1.1.1	127.254.254.254	255.0.0.0
B	128.1.1.1	191.254.254.254	255.255.0.0
C	192.1.1.1	223.254.254.254	255.255.255.0
D	224.1.1.1	239.254.254.254	
E	240.1.1.1	254.254.254.254	

Question 4

```
Carte Ethernet Ethernet 3 :  
Suffixe DNS propre à la connexion. . . : lerb.polyntl.ca  
Adresse IPv6 de liaison locale. . . : fe80::10be:6928%17  
Adresse IPv4. . . : 132.207.29.109  
Masque de sous-réseau. . . : 255.255.255.0  
Passerelle par défaut. . . : 132.207.29.1
```

Cette adresse appartient à la classe B.

Question 5

A)

Adresse IP : 1000 0100 1100 1111 0001 1101 0110 1101

Masque sous-réseau : 1111 1111 1111 1111 1111 1111 0000 0000

B)

On applique un « et logique » 1000 0100 1100 1111 0001 1101 0000 0000

Question 6

```
getaddrinfo a reussi!  
Adresse # 1  
  Flags: 0x0  
  Famille: AF_INET (IPv4)  
  Adresse IPv4: 132.207.29.103  
  Taille de cette adresse: 16 octets  
  Nom canonique: <null>  
Adresse # 2  
  Flags: 0x0  
  Famille: AF_INET6 (IPv6)  
  Adresse IPv6: 2002:84cf:1d67::84cf:1d67  
  Taille de cette adresse: 28 octets  
  Nom canonique: <null>  
Appuyer une touche pour finir...
```

Question 7

Oui, en observant le 3^e octet de l'adresse IP, en l'occurrence « 29 », on peut remarquer que les deux machines font partie du même sous-réseau.

Pour le poste L-4708-03 :

Adresse IP : 1000 0100 1100 1111 0001 1101 0110 0111

Masque sous-réseau : 1111 1111 1111 1111 1111 1111 0000 0000

On applique un « et logique » 1000 0100 1100 1111 0001 1101 0000 0000

Ce qui donne le même sous réseau que notre poste.

Question 8

La fonction `getaddrinfo` renvoie une structure de type `addrinfo` dans un pointeur passé en paramètre. Cette structure contient entre autre le type d'adresse (par exemple IPv4, IPv6 ou Bluetooth) et la longueur de cette adresse.

Question 9

5000

```
// On indique le nom et le port du serveur auquel on veut se connecter
//char *host = "L4708-XX";
//char *host = "L4708-XX.lerb.polymtl.ca";
//char *host = "add_IP locale";
char *host = "132.207.29.XXX";
char *port = "5000";
```

Question 10

```
// On va créer le socket pour communiquer avec le serveur
leSocket = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
if (leSocket == INVALID_SOCKET) {
    printf("Erreur de socket(): %ld\n\n", WSAGetLastError());
    freeaddrinfo(result);
    WSACleanup();
    printf("Appuyez une touche pour finir\n");
    getchar();
    return 1;
}
```

AF_INET

2

The Internet Protocol version 4 (IPv4) address family.

AF_INET donc IPv4.

Question 11

Le type est SOCK_STREAM qui permet une connexion bidirectionnelle. Ce type de socket utilise TCP pour les adresses de type IPv4 et IPv6.

Question 11.1

The screenshot shows two terminal windows side-by-side. The left window, titled 'X:\Documents\inf3405\tp3\Lab3_Code_Source_E2016\Lab3Serveur', shows the server's output: 'Adresse locale trouvee 132.207.29.109 :', 'En attente des connexions des clients sur le port 5000...', 'Connection acceptee De : 132.207.29.109:8463.', 'Received 7 bytes from client.', and 'Received guitare from client.'. The right window, titled 'X:\Documents\inf3405\tp3\Lab3_Code_Source_E2016\Lab3_Clients\Lab302\De...', shows the client's output: 'Adresse trouvee pour le serveur 132.207.29.109 : 132.207.29.109', 'Tentative de connexion au serveur 132.207.29.109 avec le port 5000', 'Connecte au serveur 132.207.29.109:5000', 'Saisir un mot de 7 lettres pour envoyer au serveur: guitare', 'Nombre d'octets envoyes : 7', 'Nombre d'octets recus: 7', 'Le mot reçu est GUITARE', and 'Appuyez une touche pour finir'.

Nous avons envoyé « guitare » en minuscules et avons reçu « GUITARE » en majuscules.

Question 12

```
//-----  
// Envoyer le mot au serveur  
iResult = send(leSocket, motEnvoye, 7, 0 );  
if (iResult == SOCKET_ERROR) {  
    printf("Erreur du send: %d\n", WSAGetLastError());  
    closesocket(leSocket);  
    WSACleanup();  
    printf("Appuyez une touche pour finir\n");  
    getchar();  
    return 1;  
}
```

C'est la fonction « send ».

Question 13

```
//-----  
// Maintenant, on va recevoir l' information envoyée par le serveur  
iResult = recv(leSocket, motRecu, 7, 0);  
if (iResult > 0) {  
    printf("Nombre d'octets recus: %d\n", iResult);  
    motRecu[iResult] = '\0';  
    printf("Le mot reçu est %s\n", iResult, motRecu);  
}  
else {  
    printf("Erreur de reception : %d\n", WSAGetLastError());  
}
```

C'est la fonction « recv ».

Question 14

Tel que dit à la question 11.1, le serveur renvoie le mot envoyé en majuscules.

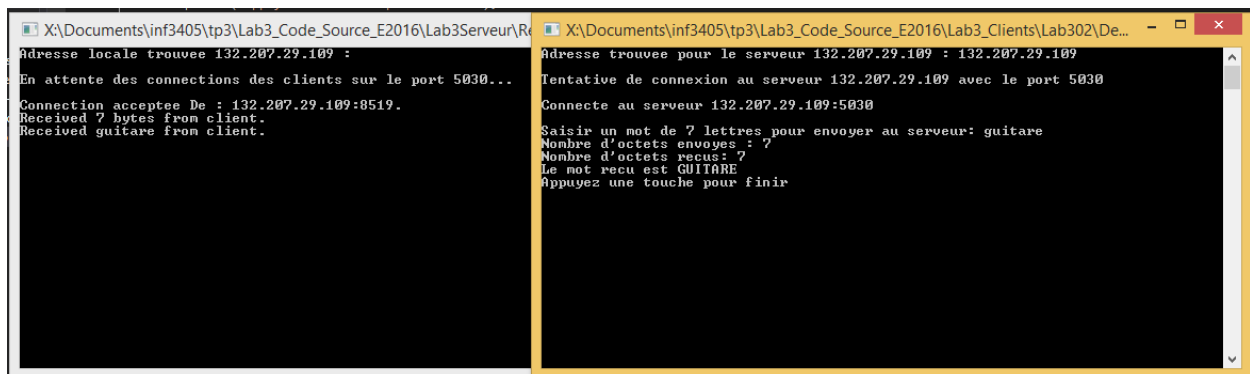
Question 15



```
X:\Documents\inf3405\tp3\Lab3_Code_Source_E2016\Lab3_Clients\Lab302\De...
Adresse trouvee pour le serveur 132.207.29.109 : 132.207.29.109
Tentative de connexion au serveur 132.207.29.109 avec le port 5030
Impossible de se connecter au serveur 132.207.29.109 sur le port 5030
Appuyez une touche pour finir
```

On ne peut pas inscrire un mot de 7 lettres puisque la connexion avec le serveur n'a pas pu être établie. En effet, il faudrait que ces derniers soient connectés sur le même port pour établir une connexion.

Question 16

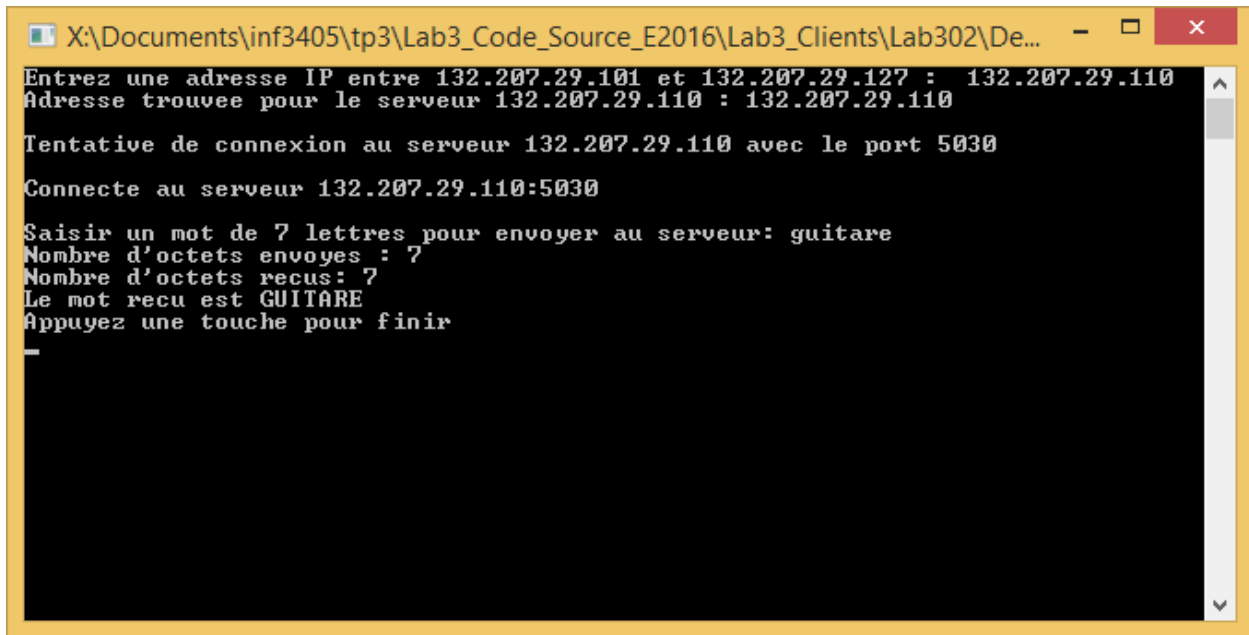


```
X:\Documents\inf3405\tp3\Lab3_Code_Source_E2016\Lab3Serveur\Re...
Adresse locale trouvee 132.207.29.109 :
En attente des connexions des clients sur le port 5030...
Connection acceptee De : 132.207.29.109:8519.
Received 7 bytes from client.
Received guitare from client.

X:\Documents\inf3405\tp3\Lab3_Code_Source_E2016\Lab3_Clients\Lab302\De...
Adresse trouvee pour le serveur 132.207.29.109 : 132.207.29.109
Tentative de connexion au serveur 132.207.29.109 avec le port 5030
Connecte au serveur 132.207.29.109:5030
Saisir un mot de 7 lettres pour envoyer au serveur: guitare
Nombre d'octets envoyes : 7
Nombre d'octets recus: 7
Le mot reçu est GUITARE
Appuyez une touche pour finir
```

L'utilité d'un port est de fournir un tube de communication entre deux machines et de fournir une distinction entre les diverses applications. Ainsi, on peut envoyer des données à plusieurs programmes pour une même adresse IP.

Question 17

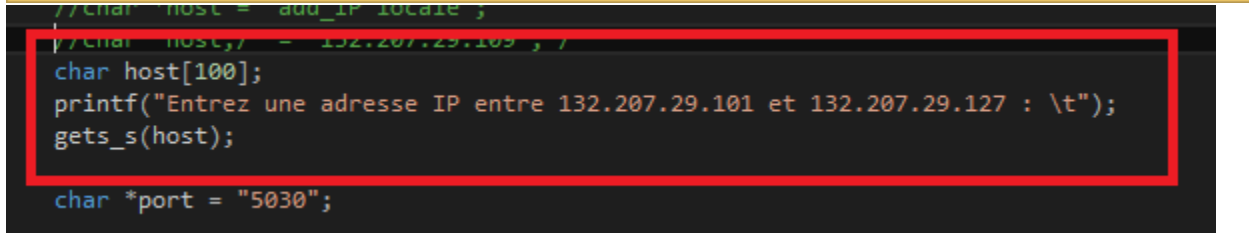


```
X:\Documents\inf3405\tp3\Lab3_Code_Source_E2016\Lab3_Clients\Lab302\De... - □ ×
Entrez une adresse IP entre 132.207.29.101 et 132.207.29.127 : 132.207.29.110
Adresse trouvee pour le serveur 132.207.29.110 : 132.207.29.110

Tentative de connexion au serveur 132.207.29.110 avec le port 5030

Connecte au serveur 132.207.29.110:5030

Saisir un mot de 7 lettres pour envoyer au serveur: guitare
Nombre d'octets envoyes : 7
Nombre d'octets recus: 7
Le mot reçu est GUITARE
Appuyez une touche pour finir
-
```



```
//char host = "add_ip_locale";
//char host, port = "132.207.29.109", "5030";
char host[100];
printf("Entrez une adresse IP entre 132.207.29.101 et 132.207.29.127 : \t");
gets_s(host);

char *port = "5030";
```

Question 18

Un thread principal exécute une boucle infinie qui reçoit les connexions entrantes. Dès qu'une connexion valide survient, un socket est créé et un nouveau thread exécutant « EchoHandler » est lancé.

Question 19

Le serveur utilise des threads Windows. Un thread par connexion. Chaque thread exécute « EchoHandler » qui, elle-même, appelle « DoSomething ».