

## Programmation Réseaux - Contrôle Continu 2

Durée : 45 minutes  
Aucun document n'est autorisé

1. De quel type sont les adresses IP codées sur 4 octets ?

Réponse :

2. Écrivez 258 sur deux octets en *little endian*.

Réponse :

3. Que renvoie `htons(258)` si la machine sur laquelle on fait l'appel utilise un codage *big endian* ?

Réponse :

4. Si en C on crée une socket de la forme `int sock=socket(PF_INET,SOCK_DGRAM,0)`, quel est le mode de communication que l'on utilise ?

Réponse :

5. Quelle commande lancer depuis le terminal pour écouter sur le port 2424 en mode paquet et afficher ce que l'on reçoit ?

Réponse :

6. Deux méthodes `synchronized` peuvent-elles s'exécuter en même temps ?

Réponse :

7. Quelle est la valeur de l'entier renvoyé par la fonction fork si l'on est dans le processus appelant ?

Réponse :

8. Quel est le problème principal du morceau de code suivant :

```
int sock=socket(PF_INET,SOCK_STREAM,0);
struct sockaddr_in address_sock;
address_sock.sin_family=AF_INET;
address_sock.sin_port=htons(4242);
int r=connect(sock,(struct sockaddr *)&address_sock,
             sizeof(struct sockaddr_in));
```

Réponse :

9. Quel est le problème du morceau de code suivant :

```
DatagramSocket dso=new DatagramSocket();
byte [] data;
String s="MESSAGE";
data=s.getBytes();
DatagramPacket paquet=new DatagramPacket(data,data.length);
dso.send(paquet);
```

Réponse :

10. Qu'affiche le programme suivant :

```
int main(){
    int a=3;
    int pid = fork();
    if(pid==0){
        a++;
    }
    printf("a vaut : %d\n",a);
    return 0;
}
```

Réponse :

11. Quelle est la valeur affichée par le programme suivant ?

```
void *incremente(void *ptr){
    int *s=(int *)ptr;
    int a>(*s)+1;
    *s=a;
}

int main(){
    int a=0;
    pthread_t th[50];
    for(int i=0;i<50;i++){
        pthread_create(&th[i],NULL,incremente,&a);
    }
    for(int i=0;i<50;i++){
        pthread_join(th[i],NULL);
    }
    printf("%d\n",a);
    return 0;
}
```

Réponse :

12. Comment recevoir et afficher le message du programme suivant (sans écrire de programme) ?

```
public class Client{
    public static void main(String [] args){
        try{
            Socket socket=new Socket("lucien",3333);
            PrintWriter pw=new PrintWriter(
                new OutputStreamWriter(
                    socket.getOutputStream()));
            pw.println("Salut!");
            pw.flush();
            pw.close();
        }
        catch(Exception e){
            System.out.println(e);
            e.printStackTrace();
        }
    }
}
```

```
}
```

Réponse :

13. On considère le code suivant :

```
int main() {
    int sock=socket(PF_INET,SOCK_DGRAM,0);
    struct sockaddr_in address_sock;
    address_sock.sin_family=AF_INET;
    address_sock.sin_port=htons(3333);
    address_sock.sin_addr.s_addr=htonl(INADDR_ANY);
    int r=bind(sock,(struct sockaddr *)&address_sock,
              sizeof(struct sockaddr_in));
    struct sockaddr_in emet;
    socklen_t a=sizeof(emet);
    if(r==0){
        char tampon[100];
        while(1){
            int rec=recvfrom(sock,tampon,100,0,
                            (struct sockaddr *)&emet,&a);
            tampon[rec]='\0';
            printf("Message_recu_:_%s\n",tampon);
        }
    }
    return 0;
}
```

Si on veut exécuter ce programme avec le programme de la question précédente pour qu'ils communiquent, sur quelle machine doit-on exécuter les deux programmes ? Quel programme doit-on "lancer" en premier pour que cela fonctionne ?

Réponse :

14. Donnez un programme Java pour recevoir et afficher le message envoyé par le programme Java de la question 12.

Réponse :

15. Donnez un programme C pour faire afficher Hello World! au programme Java suivant s'exécutant sur monjetas. On rappelle la signature de la fonction suivante :

```
ssize_t sendto(int s, const void *buf, size_t l, int flags,
const struct sockaddr *dest, socklen_t dest_len)
```

```
public class Serveur{
    public static void main(String [] args){
        try{
            DatagramSocket dso=new DatagramSocket(5252);
            byte [] data=new byte[100];
            DatagramPacket paquet=new DatagramPacket(
                data ,data.length );
            while(true){
```

```
    dso.receive(paquet);  
    System.out.println(new String(data,0,  
                                paquet.getLength()));  
    }  
  }  
}
```

Réponse :