

**NOTA :**

- tous documents autorisés
- les exercices sont tous indépendants
- Tous les programmes ou fonctions demandés seront écrits en C standard, les fautes de syntaxe bénignes ne seront pas sanctionnées.

**1 Le langage C :****1.1 Base standard: 2 points**

Écrire un programme permettant de saisir un entier (32 bits) au clavier. Le programme obligera l'opérateur à entrer un entier compris dans l'intervalle [0, 100]. Tant que l'opérateur entrera une valeur incorrecte on lui proposera d'entrer cet entier. Attention l'entier saisi par l'opérateur peut être négatif.

**1.2 Base technique: 2 points**

A l'aide des opérateurs de masquage et de décalage à gauche, écrire un programme qui affiche le plus grand nombre puissance de 2 (2,4,8,16,32,64,...) avec lequel on peut multiplier l'entier saisi à l'exercice 1.1. pour ne pas dépasser la capacité de calcul d'un entier non signé 32 bits. On rappelle qu'un décalage à gauche d'un bit correspond à une multiplication par 2.

**1.3 Une fonction: 3 points**

Écrire en C la fonction bool TestTrier (*short Val[]*, *int taille*). Cette fonction recevra en paramètre un tableau d'entiers courts signés *Val*, son nombre d'éléments *taille* et fournira en retour true si le tableau est trié du plus petit au plus grand, false dans le cas où il y a un désordre quelconque dans le tableau.

**1.4 Un tableau dynamique: 3 points**

Soit Tab un tableau d'entiers courts.

On dispose de la fonction void Trier(short \*,int) qui trie le tableau passé en premier paramètre, le deuxième paramètre est son nombre d'éléments.

Écrire un programme qui permet de:

- saisir au clavier le nombre d'éléments de ce tableau.
- créer dynamiquement ce tableau.
- saisir tous les éléments de ce tableau au clavier.
- Vérifier à l'aide de la fonction TestTrier (question 1.3) si Tab est ordonné s'il ne l'est pas on l'ordonnera avec la fonction Trier (la fonction Trier n'est pas à écrire).

**2 Réseaux Locaux Industriels****2.1 Communication Point à Point RS232: 3 points**

On veut échanger des fichiers ASCII entre différentes machines par le protocole ETX/ ACK. Le fichier est découpé en bloc de 256 octets qui sont émis dans des trames qui suivent la structure suivante:

Début	Données			CRC		FIN
STX	Octet1	.....	Octet N	PF	pf	EOT

On dispose de la fonction bool CtrlCRC(char \*Trame, int NbOctetDeTrame) qui retourne true si le CRC contenu dans la trame est bon.

En utilisant les fonctions de gestion de voie série sous Windows (en particulier la lecture et l'écriture de données), on demande de compléter le code source fourni pour que la fonction retourne dans le tableau Data les données reçues (uniquement les données), dans Nb le nombre d'octets du tableau Data.

Quel autre type de passage de paramètre aurait-on pu utiliser pour le paramètre Nb?

**2.2 Sous-Couches MAC-LLC : 3 points**

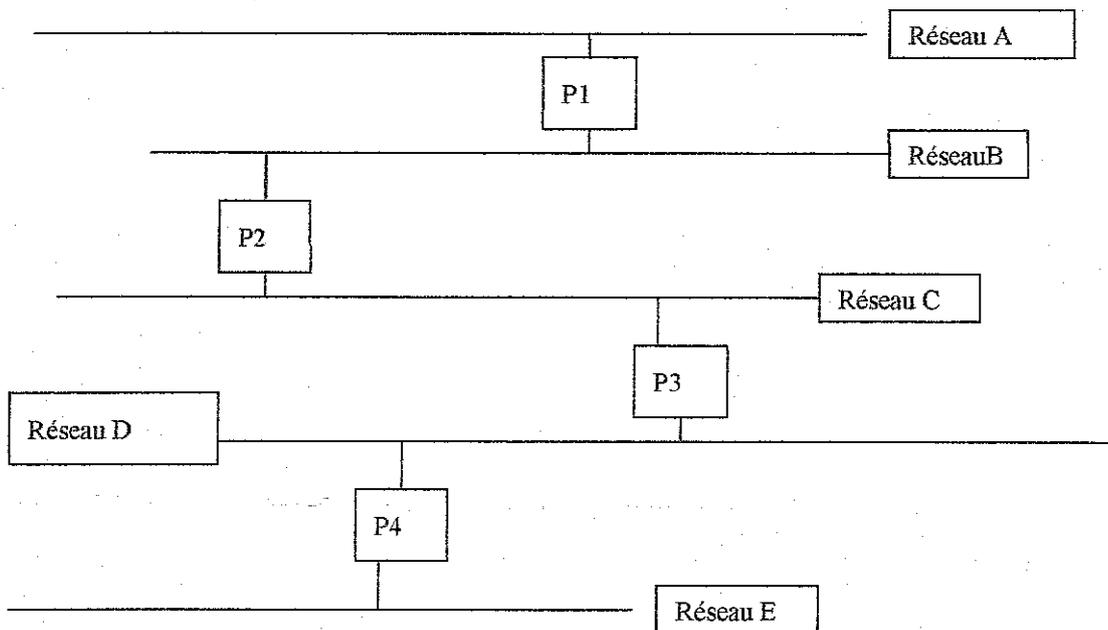
Donnez en les expliquant rapidement au moins 2 protocoles de la couche MAC et un de la couche LLC.

### 3 Plan d'adressage : 4 points

Une entreprise dispose de 5 services :

Services	Nb Postes Info	Nb Imprimantes réseau	Nb caméra IP
A: Administratif	50	3	0
B: Développement	610	2	0
C: Laboratoire d'essais	70	1	0
D: Atelier	750	0	10
E: Expédition	10	1	0

L'architecture réseau de l'entreprise est la suivante :



Choisir une adresse réseau privée IP locale et son masque pour l'entreprise. Pour chaque service on donnera :

- L'adresse de sous réseau
- Le masque réseau

Les réseaux A et E n'ont qu'une passerelle le routage ne posera pas de problème, donner son principe dans le cas où une station du réseau A envoie :

Cas 1 : un message à une autre station du même réseau.

Cas 2 : un message à une station du réseau B.

Comment sera résolu le routage sur les autres réseaux qui possèdent 2 passerelles ?

**HANDLE \*voie; // variable globale initialisée au préalable sur la voie série COM1**

```

bool Recevoir(char *Data,unsigned int &Nb)
{
    bool rts, test=false;
    int cpt=0,i;
    char NACK=0x26,ACK=0x06,EOT=0x04;
    char Trame[260];
    unsigned long NbEcrits,NbRecus;

    while ((cpt <16) && (!test))
    {
        i=0;
        do
        {
            // Lire un octet sur la voie série et le mettre dans Trame

        } while (Trame[i-1] != EOT);

        test = CtrlCRC(Trame, i-1)
        if (! test)
        { // erreur de transmission il faut écrire sur la voie série NACK

                cpt++;
            }
        else
        { // Trame reçue OK
            // il faut copier dans Data les données

        }
    }

    // il faut initialiser Nb

    //il faut écrire sur la voie série ACK

}

if (cpt ==16)
    rts= false;
else
    rts=true;
return rts;
}

```

NOM :

Signature: