

Durée : 1h. Documents limités à une page A4. Calculatrice autorisée (mémoire vidée). Aucun échange entre étudiants ne sera toléré (pas même gomme ou stylo). Téléphones ou autres appareils communicants éteints et rangés dans vos sacs. Toute réponse non justifiée (par quelques mots) n'apporte aucun point. N'oubliez pas votre nom sur la copie.

Comme vous le savez, une adresse IP est composée de 4 octets (du moins dans la norme IPv4, encore beaucoup utilisée) . Dans la machine, ils sont bien évidemment stockés en binaire. Mais pour les rendre lisibles à l'homme, on les affiche sous forme de 4 nombres décimaux séparés par un point. Et tout cela en code ASCII. Nous allons étudier ici le passage de l'un à l'autre.

- 1) Le code ASCII du point est 46 en décimal. Combien cela fait-il en hexa et en binaire ? Dites en quelques mots comment vous obtenez ces résultats (excepté calculatrice), le plus simple sera le mieux noté.
- 2) Les chiffres de 0 à 9 sont codés dans l'ordre de 48 à 57 (en décimal). Donnez leur codage en hexa et en binaire, dans un tableau (commencez par le code du 0, les autres en découleront facilement). Attention laissez de la place à droite pour une colonne supplémentaire.
- 3) Pour transformer un chiffre entre 0 et 9 ASCII en sa valeur binaire (afin de pouvoir faire des calculs), que suffit-il de faire ? (en une phrase, d'après ce que vous venez de remarquer). Vous pouvez utiliser des opérations booléennes.
- 4) Rajoutez une colonne au tableau précédent, notez-y ce que vaut en binaire dix fois le nombre (ligne 0 : zéro, ligne 1 : dix, ligne 2 : vingt,...)
- 5) Transformez les deux nombres binaires suivants en décimal : $C=1100100_b$ et $D=11001000_b$
- 6) Quand un nombre décimal (entre 0 et 255) est fourni en ASCII, pour le transformer en sa valeur binaire sur 8 bits, il faut :
 - a) isoler les unités, dizaines et centaines,
 - b) les transformer en leur valeur binaire (question 3),
 - c) ajouter l'unité à dix fois la dizaine (calculs en binaire) (question 4)
 - d) suivant la valeur de la centaine, ajouter (toujours en binaire) cent ou deux cents si nécessaire.

Quelle opération permet d'isoler les unités ? comment écrit-on cela en C ?

- 7) Questions générales :
 - a) Dans un système d'exploitation, les droits d'accès (aux fichiers par exemple) dépendent de l'UID et du GID. Qu'est-ce que l'UID et le GID ?
 - b) Quelle différence y a-t-il en un système multi-tâche et un système temps réel ? D'ailleurs, qu'est-ce que le « système » dont on parle dans ces deux expressions ?
 - c) Quelles différences y a-t-il entre le langage machine et un langage « évolué » (3^e génération par exemple) ?
 - d) Qu'est-ce que le code GRAY, et surtout, pourquoi l'utilise-t-on ?
- 8) Question bonus : ceux qui ont fini, ont presque tout juste mais veulent un 20 peuvent donner la suite d'instructions en C réalisant l'algorithme précisé dans la question 6. On part d'un entier 32 bits contenant les 3 codes ASCII (donc 24 bits utiles, les 8 autres étant à 0), à la fin on obtient la valeur exacte (dans un autre entier, qui lui n'aura que 8 bits utilisés sur les 32). « Suite d'instruction » signifie pour moi qu'il n'y a pas besoin de prévoir de #include, de « main », de scanf.... juste la suite d'instructions réalisant le calcul.