

---

## Correction

---

EIDGENÖSSISCHE TECHNISCHE HOCHSCHULE – LAUSANNE  
POLITECNICO FEDERALE – LOSANNA  
SWISS FEDERAL INSTITUTE OF TECHNOLOGY – LAUSANNE

Faculté Informatique et Communications  
Cours ICC aux sections MA et PH  
Chappelier J.-C.



# INFORMATIQUE, CALCUL & COMMUNICATIONS

## Sections MA & PH

### Correction Examen intermédiaire III

16 décembre 2016

#### SUJET 1

#### Instructions :

- Vous disposez d'une heure quinze minutes pour faire cet examen (15h15 - 16h30).
- L'examen est composé de 2 parties : un questionnaire à choix multiples, à 12 points, prévu sur 35 minutes, et une partie à questions ouvertes, à 8 points, prévue sur 40 minutes. Mais vous êtes libres de gérer votre temps comme bon vous semble.
- **AUCUN DOCUMENT N'EST AUTORISÉ, NI AUCUN MATÉRIEL ÉLECTRONIQUE.**
- Pour la première partie (questions à choix multiples), chaque question n'a qu'une seule réponse correcte parmi les quatre propositions. Indiquez vos réponses en bas de **cette** page en écrivant *clairement* pour chaque question une lettre majuscule parmi A, B, C et D. (Vous êtes autorisés à dégrafer cette page) **Aucune autre réponse ne sera considérée**, et en cas de rature, ou de toute ambiguïté de réponse, nous compterons la réponse comme fausse.
- Pour la seconde partie, répondez directement sur la donnée, à la place libre prévue à cet effet. Aucune feuille supplémentaire ne sera considérée.
- Toutes les questions comptent pour la note finale.

#### Réponses aux quiz :

Reportez ici *en majuscule* la lettre de la réponse choisie pour chaque question, sans aucune rature.

1	2	3	4	5	6	7	8	9	10	11	12
C	C	D	C	A	B	D	A	B	B	C	C

## PARTIE QUIZ

**Question 1)** Si un réseau TCP/IP transmet des paquets contenant 16 Kio de données au rythme de 32 paquets par seconde, mais perd un paquet sur quatre, combien de temps prendra le transfert d'un fichier de 2 Mio en négligeant tous les autres temps que ceux de transmission de paquets ?

- A] environ 3 s  
 B] environ 4 s  
 ✓C] environ 5 s (mais moins que 5.35 s)  
 D] au moins 5.35 s

**Question 2)** Laquelle de ces propriétés est implémentée au niveau de la couche « transport » d'un réseau Internet ?

- A] Transmission de photons par fibre optique.  
 B] Interprétation du contenu des emails.  
 ✓C] Contrôle de l'intégrité des données.  
 D] Calcul distribué du plus court chemin pour délivrer les paquets.

**Question 3)** L'entête d'un paquet réseau sert :

- A] uniquement à stocker de l'information redondante en cas de perte.  
 B] de clé de chiffrement pour une communication sécurisée.  
 C] uniquement à faire le tri des différents paquets arrivant sur le réseau.  
 ✓D] à transmettre des métadonnées sur le paquet.

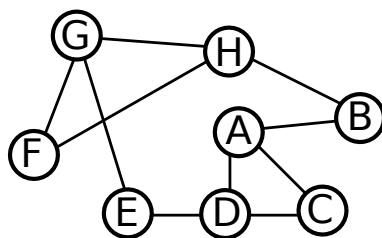
**Question 4)** Considérez la liste de moyens de stockage suivante :

**RAM, Flash, disque dur, bande magnétique.**

Pour chacune des propriétés suivantes : **latence**, **prix** (par octet), **capacité**, la liste ci-dessus est-elle classée (en 2016) par ordre croissant ou décroissant de valeur ?

- A] latence : décroissant ; prix : décroissant ; capacité : décroissant.  
 B] latence : croissant ; prix : croissant ; capacité : croissant.  
 ✓C] latence : croissant ; prix : décroissant ; capacité : croissant.  
 D] latence : décroissant ; prix : croissant ; capacité : décroissant.

**Question 5)** Laquelle des lignes suivantes fait partie de la table de routage de A pour le réseau suivant (en exprimant les distances en nombre d'arcs) :



- ✓A] 

dest.	dir.	dist.
F	B	3

 B] 

dest.	dir.	dist.
H	D	4

 C] 

dest.	dir.	dist.
D	G	2

 D] 

dest.	dir.	dist.
A	F	3

suite au dos ➞

**Question 6)** Une machine utilise des registres de 32 bits. L'entropie (telle que définie en cours) de la séquence de 0/1 contenue dans le registre `r1` peut-elle changer si on : (1) inverse tous ses bits ; (2) on exécute l'instruction « somme `r1`, `r1`, 1 » ?

- A] (1) oui ; (2) oui.     B] (1) non ; (2) oui.    C] (1) oui ; (2) non.    D] (1) non ; (2) non.

**Question 7)** On considère les deux problèmes suivants :

- chercher la distance minimale entre chaque paire de nœuds dans un réseau IP ;
- cherche un chemin qui passe par tous les nœuds d'un réseau IP et dont la distance (totale) est minimale.

Que peut-on en dire ?

- A] Aucun de ces problèmes n'est dans P et les deux sont dans NP.  
 B] Aucun de ces problèmes n'est dans P et l'un des deux est dans NP.  
 C] Un de ces problèmes est dans P et un est dans NP.  
 D] Un de ces problèmes est dans P et les deux sont dans NP.

**Question 8)** Pour calculer la somme des éléments d'une liste de  $n$  nombres, on la sépare en  $k$  sous-listes dont on calcule *en parallèle* chacune des sommes de leurs éléments, puis on termine en sommant toutes ces sommes partielles.

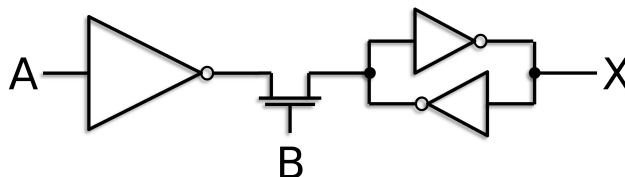
Si l'on note  $a$  le nombre total d'additions effectuées,  $t$  le temps total (en ms) pour avoir le résultat final et  $m = \lceil n/k \rceil$  la taille (maximale) d'une sous-liste, que valent  $a$  et  $t$  en supposant qu'une addition prend 1 ms ?

- A]  $a = n - 1$     et     $t = m + k - 2$        C]  $a = n - 1$     et     $t = n - 1$   
 B]  $a = m + k - 2$  et     $t = m + k - 2$        D]  $a = n + k - 2$  et     $t = (m - 1) \times k$

**Question 9)** Un processeur fait des additions de nombres entiers signés représentés sur 8 bits. Si ses registres `r1`, `r2` et `r3` contiennent respectivement 10101101, 11010110 et 01010101, que contient `r3` après l'instruction « somme `r3`, `r1`, `r2` » ?

- A] 11011000     B] 10000011    C] 01111100    D] 00000011

**Question 10)** On considère le système logique suivant :



auquel on soumet les entrées A et B suivantes à quatre instants consécutifs :

t	A	B	X
1	0	1	$x_1$
2	1	0	$x_2$
3	0	1	$x_3$
4	1	1	$x_4$

Quelles sont les quatre sorties ( $x_1, x_2, x_3, x_4$ ) correspondantes ?

- A] (1, 0, 1, 0)     B] (0, 0, 0, 1)    C] (0, 1, 0, 1)    D] (1, 1, 1, 0)

**Question 11)** Quelle est la table de vérité du programme ci-contre (sachant que r1 et r2 contiennent soit 0 soit 1) ?

A] 

r1	r2	r3
0	0	0
1	0	0
0	1	0
1	1	10

B] 

r1	r2	r3
0	0	0
1	0	1
0	1	1
1	1	1

C] ✓ 

r1	r2	r3
0	0	0
1	0	1
0	1	1
1	1	0

D] Aucune des trois.

```

1: cont_egal r1, 0, 5
2: cont_egal r2, 0, 5
3: charge r3, 0
4: continue 6
5: somme r3 r1 r2
6: // fin (stop)

```

**Question 12)** Les signaux  $X_1(t) = 3 \sin(\frac{\pi t}{2})$  et  $X_2(t) = 3 \sin(2\pi t + \frac{\pi}{3})$  sont filtrés par un filtre à moyenne mobile de période d'intégration  $T_c = 8$  secondes.

Que peut on dire des amplitudes  $A_1$  et  $A_2$  de ces signaux après filtrage ?

- A]  $A_1 > A_2$       B]  $A_1 < A_2$       ✓C]  $A_1 = A_2 = 0$       D]  $A_1 < 1/8$  et  $A_2 > 1/8$

suite au dos 

## PARTIE EXERCICES

### 1 – Inverse du compilateur [4 points]

Considérons le code assembleur suivant :

```

1: cont_egal r1, r2, 7
2: cont_pp   r1, r2, 5
3: soustrait r1, r1, r2
4: continue 1
5: soustrait r2, r2, r1
6: continue 1
7: // fin (stop)

```

**Question 13)** [1 point] Que vaut  $r1$  lorsque le programme ci-dessus se termine, si au départ  $r1$  contenait 01000110 et  $r2$  00101010 ? Justifiez votre réponse.

Réponse : 14 (i.e. 00001110) :  $70(r1) - 42(r2) = 28; 42 - 28 = 14; 28 - 14 = 14$ .

**Question 14)** [1 point] Ecrivez l'algorithme correspondant au programme ci-dessus.

Réponse :

<b>PGCD</b>
entrée : Deux nombres entiers (strictement positifs) $r1$ et $r2$ sortie : $PGCD(r1, r2)$
<p><b>Si</b> <math>r1 = r2</math>            <b>Sortir</b> : <math>r1</math>  <b>Sinon</b>, si <math>r1 &gt; r2</math>            <b>Sortir</b> : <math>PGCD(r1 - r2, r2)</math>            <b>Sortir</b> : <math>PGCD(r1, r2 - r1)</math></p>

ou

<b>PGCD</b>
entrée : Deux nombres entiers (strictement positifs) $r1$ et $r2$ sortie : $PGCD(r1, r2)$
<p><b>Tant que</b> <math>r1 \neq r2</math>            <b>Si</b> <math>r1 &gt; r2</math>                <math>r1 \leftarrow r1 - r2</math>            <b>Sinon</b>                <math>r2 \leftarrow r2 - r1</math>  <b>Sortir</b> : <math>r1</math></p>

**Question 15)** [1 point] En une phrase, que fait cet algorithme (mathématiquement) ?

**Réponse :** PGCD( $r_1$ ,  $r_2$ )

**Question 16)** [1 point] Quelle est la complexité (temporelle pire cas) de cet algorithme ?

**Réponse :** Le pire cas est quand l'un des deux vaut 1 et l'autre  $n$ . L'algorithme fait alors  $n - 1$  soustractions (et tests). Si l'on néglige la complexité de la soustraction, alors la complexité est en  $\mathcal{O}(\max(r_1, r_2))$ .

---

## 2 – La GameBoy de grand-maman sur mon ordi [4 points]

Il est possible de jouer aux anciens jeux vidéo sur des ordinateurs modernes grâce à une technique appelée « *émulation* » : le processeur pour lequel le jeu a été écrit est complètement simulé par un programme sur la machine moderne, qui devient ainsi capable d'interpréter les instructions avec lesquelles le vieux jeu a été écrit (et que le processeur moderne ne saurait pas interpréter directement).

Considérons un ancien processeur simplifié avec uniquement 4 registres de 8 bits et uniquement 3 instructions, prenant chacune trois opérandes sur 8 bits :

- « `soustr_v N, valeur, M` » qui met dans le registre numéro  $N$  le résultat de la soustraction de la valeur du registre numéro  $M$  à la valeur donnée (sur 8 bits) :  $rN \leftarrow \text{valeur} - rM$
- « `soustr_r N, L, M` » qui met dans le registre numéro  $N$  le résultat de la soustraction des valeurs des registres numéros  $L$  et  $M$  :  $rN \leftarrow rL - rM$
- « `cont_pp N, M, L` » qui saute à la ligne  $L$  si la valeur du registre numéro  $N$  est strictement inférieure à celle du registre numéro  $M$ .

**Note :**  $L$  étant représenté sur 8 bits, les programmes n'ont pas plus de 256 lignes.

**Question 17) [0.5 point]** En supposant que tous les registres sont à zéro au départ, que contient le registre numéro 2 à la fin du programme ci-dessous. Justifiez votre réponse.

```
1: soustr_v 1, 5, 2
2: soustr_v 2, 7, 3
3: soustr_v 3, 26, 3
4: soustr_v 1, 0, 1
5: soustr_r 2, 2, 1
6: cont_pp 2, 3, 5
```

Réponse :			
ligne	r1	r2	r3
1	$5 - 0 = 5$	0	0
2	5	$7 - 0 = 7$	0
3	5	7	$26 - 0 = 26$
4	$0 - 5 = -5$	7	26
5	-5	$7 - (-5) = 12$	26
5	-5	$12 - (-5) = 17$	26
5	-5	$17 - (-5) = 22$	26
5	-5	$22 - (-5) = 27$	26

**Question 18) [1.5 point]** Nous voulons maintenant écrire un émulateur de ce vieux processeur.

Complétez l'algorithme ci-contre (page suivante) de sorte qu'il émule le processeur précédent (c.-à-d. simule son fonctionnement). Cet algorithme prend en entrée une liste d'instructions du (vieux) processeur, dans laquelle chaque instruction est représentée comme un quadruplet (instruction, opérande 1, opérande 2, opérande 3).

Emulateur
entrée : liste $L$ d'instructions (quadruplets)
sortie : état des registres
<pre> <math>n \leftarrow \text{taille}(L)</math> ligne <math>\leftarrow 1</math> registres <math>\leftarrow (0, 0, 0, 0)</math> <b>Tant que</b> ligne <math>\leq n</math>   (instr, <math>a</math>, <math>b</math>, <math>c</math>) <math>\leftarrow L(\text{ligne})</math>   ligne <math>\leftarrow \text{ligne} + 1</math>   <b>Si</b> instr = soustr_v     registres(<math>a</math>) <math>\leftarrow b - \text{registres}(c)</math>   <b>Sinon, si</b> instr = soustr_r     registres(<math>a</math>) <math>\leftarrow \text{registres}(b) - \text{registres}(c)</math>   <b>Sinon, si</b> registres(<math>a</math>) &lt; registres(<math>b</math>)     ligne <math>\leftarrow c</math> <b>Sortir</b> : registres </pre>

**Question 19)** [1 point] Pour simplifier l'explication, nous avons représenté chaque instruction avec trois opérandes sur 8 bits chacun. Mais en réalité toute cette information (une instruction et ses opérandes) est représentée avec moins de bits. Combien de bits faut-il au minimum ? Expliquez votre codage puis réécrivez uniquement en binaire les lignes 1, 5 et 6 du programme de la question 17.

**Réponse :** 2 bits sont nécessaires et suffisant pour représenter les trois instructions, et 2 bits aussi pour les quatre registres.

soustr\_v nécessite donc au moins  $2 + 2 + 8$  (valeur)  $+ 2 = 14$  bits, qui doivent être alignés sur des mots de 8 bits : donc 2 mots de 8 bits.

soustr\_r nécessite au moins  $2 + 2 + 2 + 2 = 8$  bits ; donc 1 mot.

cont\_pp nécessite au moins  $2 + 2 + 2 + 8$  (numéro de lignes)  $= 14$  bits ; donc 2 mots.

Par exemple :

- soustr\_v codé 00, soustr\_r codé 01 et cont\_pp codé 10 ;
- registres codé comme « numéro moins 1 » ;
- le tout écrit dans l'ordre de lecture et alignés à droite pour les 14 vers 16 bits (i.e. codé sur les bits de poids fort) (ce qui n'est pas le plus malin car coupe en deux la valeur de soustr\_v et le numéro de ligne de cont\_pp ; mais bon, c'est juste un exemple simple...)

donne :

1: 00 00 00000110 01 = 00000000 01100100

5: 01 01 01 00 = 01010100

6: 10 01 10 00000101 = 10011000 00010100

**Question 20)** [1 point] Le modèle simplifié d'ancien processeur présenté au début est-il capable de simuler (ou « émuler ») une machine de Turing universelle ? Justifiez.

**Réponse :** non, essentiellement parce qu'il n'y a pas de mémoire, pas « d'entrées/sorties » (simulation du ruban infini).

La limitation du jeu d'instructions n'est pas un bon argument car on peut en fait effectuer beaucoup d'opérations plus complexes telles que charger une valeur dans un registre, faire une somme, etc. en les écrivant à l'aide des trois instructions de base. On peut donc imaginer stocker la position de la tête dans un registre, et pouvoir la « déplacer » (ajout/soustraction de 1) ; charger la valeur lue dans un autre registre, mémoriser l'état dans un troisième.