

NOM : Hanon Ymous  
(000000)  
Place : 0

#0000



## Information, Calcul et Communication (SMA/SPH) : Examen final

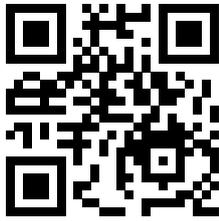
22 décembre 2022

### SUJET 1

#### INSTRUCTIONS (à lire attentivement)

**IMPORTANT!** Veuillez suivre les instructions suivantes à la lettre sous peine de voir votre examen annulé dans le cas contraire.

1. Vous disposez de deux heures quarante-cinq minutes pour faire cet examen (8h15 – 11h00).
2. Vous devez **écrire à l'encre noire ou bleu foncée**, pas de crayon ni d'autre couleur.  
N'utilisez **pas non plus de stylo effaçable** (perte de l'information à la chaleur).
3. Vous avez droit à toute documentation papier.  
En revanche, vous ne pouvez pas utiliser d'ordinateur personnel, ni de téléphone portable, ni aucun autre matériel électronique.
4. Répondez aux questions directement sur la donnée, **MAIS** ne mélangez pas les réponses de différentes questions!  
Ne joignez aucune feuilles supplémentaires; **seul ce document sera corrigé**.
5. Lisez attentivement et *complètement* les questions de façon à ne faire que ce qui vous est demandé. Si l'énoncé ne vous paraît pas clair, ou si vous avez un doute, demandez des précisions à l'un des assistants.
6. L'examen comporte 7 exercices indépendants sur 20 pages, qui peuvent être traités dans n'importe quel ordre, mais qui ne rapportent pas la même chose (les points sont indiqués, le total est de 125 points).  
Tous les exercices comptent pour la note finale.



## Question 1 – Quelques questions [22 points]

① [4 points] Considérons une séquence de lettres  $X$  dont l'entropie est de 3.25 bit. Pour *chacune* des valeurs suivantes (en bits) : 3.45, 3.15, 3.25, 4.35 :

- (a) *peut*-il exister un code binaire sans-préfixe et sans perte de  $X$  ayant une telle longueur moyenne ? (oui ou non)
- (b) pensez-vous qu'il existe un code (sans-préfixe et sans perte) pour  $X$  ayant une longueur moyenne strictement plus petite ? (oui, non ou peut-être)

Répondez dans le tableau ci-dessous. **Attention !** il y aura des pénalités pour les réponses fausses.

longueur moyenne	(a) peut exister ?	(b) existe-t-il un meilleur code ?
3.45		
3.15		
3.25		
4.35		

② [3 points] En utilisant RSA, vous souhaitez signer<sup>1</sup> une information que vous envoyez à un ami dont la clé publique est  $(47, 377)$ . Votre clé publique est  $(101, 247)$  et votre clé privée est 77. Votre ami, dont la clé privée est 143, déchiffre le message qu'il a reçu et trouve 10100111.

Quelle est la signature que vous avez envoyée ?

Exprimez votre réponse sous la forme «  $x^y \bmod z$  » et justifiez pleinement votre réponse.

**Réponse et justification :**

1. uniquement signer, pas protéger



Considérez le code assembleur suivant :

```
1: cont_ppe r2 0 9
2: multiplie r3 4 r0
3: charge r0 r1
4: multiplie r1 -5 r1
5: somme r1 r1 r3
6: somme r1 r1 7
7: somme r2 r2 -1
8: continue 1
9: rien (fin)
```

③ [3 points] Si  $r_0$  contient 2,  $r_1$  contient 1, et  $r_2$  contient 3, que contient  $r_0$  à la fin du programme ?

Justifiez brièvement votre réponse.

**Réponse et justification :**

④ [3 points] Ecrivez une fonction C++ *réursive* d'au plus deux lignes<sup>2</sup> dont la compilation pourrait donner le code assembleur ci-dessus.

**Réponse :**

⑤ [2 points] Quelle est la complexité de l'algorithme correspondant ?

Justifiez brièvement votre réponse.

**Réponse et justification :**

suite au dos 

2. c.-à-d. dont le corps contient au maximum deux ; (points-virgules)



⑥ [3 points] En supposant que **taille()** est en  $\Theta(1)$ , quelle est la complexité de l'algorithme ci-contre, où  $\lfloor x \rfloor$  représente la partie entière inférieure de  $x$  et  $L[i : j]$  représente la sous-liste  $(L(i), \dots, L(j))$  si  $i \leq j$  ou la liste vide si  $i > j$ ? Justifiez votre réponse.

Réponse et justification :

<b>algo1</b>
entrée : <i>une liste L non vide</i> sortie : ??
$n \leftarrow \mathbf{taille}(L)$ <b>Si</b> $n = 1$   <b>Sortir</b> : $L(1)$ $k \leftarrow \lfloor \frac{n}{3} \rfloor$ <b>Si</b> $k = 0$   $k \leftarrow 1$ $p \leftarrow \mathbf{algo1}(L[1 : k])$ $q \leftarrow \mathbf{algo1}(L[k + 1 : n])$ <b>Si</b> $p < q$   <b>Sortir</b> : $p$ <b>Sortir</b> : $q$

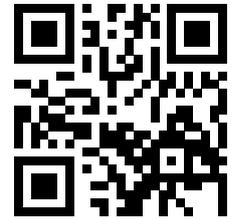
⑦ [4 points] Le plan du marché de Noël comporte 8 stands de nourriture, 3 stands d'habits chauds, 6 stands de bijoux et 1 stand pour faire des dons à une association caritative.

Quelle est l'entropie du plan des stands du marché de Noël?

Donnez votre réponse sous la forme  $a + b \log_2(3)$ , avec  $a$  et  $b$  deux fractions (non nulles) ; puis *justifiez* la brièvement.

Réponse et justification :

Ne pas écrire dans cette zone.



---

## Question 2 – Course à pied [20 points]

Vous organisez une course à pied et vous voulez archiver les temps de la façon la plus compacte possible. Ces temps sont représentés par une valeur en minutes et une valeur en secondes ; p.ex. 30 minutes et 58 secondes. Toutes les minutes entre 26 et 33 sont présentes (huit valeurs possibles) et toutes les secondes entre 00 et 59 aussi.

① Un stagiaire a proposé un premier codage consistant à simplement représenter la valeur entière positive des minutes, suivie de celle des secondes, *chacune* sur le nombre minimum de bits pour celle-ci.

a) [2 points] Combien de bits cette représentation utilise-t-elle pour chaque temps ?

**Justifiez** votre réponse.

b) [2 points] Concrètement comment est codé le temps 30 minutes et 58 secondes ?

**Réponses et justification :**

② [1 point] Une seconde idée suggérée a été de coder les minutes, toujours séparément des secondes, mais sur le nombre minimum de bits pour coder les valeurs de minutes utilisées.

Combien de bits cette représentation utilise-t-elle pour chaque temps ?

**Justifiez** votre réponse.

**Réponse et justification :**

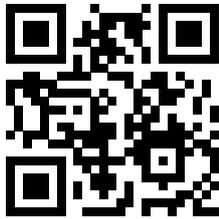
③ [2 points] Une troisième idée proposée consiste à coder les minutes et les secondes ensembles sur le nombre minimum de bits pour coder toutes les valeurs de temps possibles.

Combien de bits cette représentation utilise-t-elle pour chaque temps ?

**Justifiez** votre réponse.

**Réponse et justification :**

suite au dos 



④ Finalement, vous prenez en charge ce codage et commencez par constater que les secondes sont uniformément distribuées, mais que les minutes ne le sont pas du tout. Sur les statistiques du passé, vous constatez les comptes suivants :

26	27	28	29	30	31	32	33
2	2	16	16	64	16	10	2

- a) [6 points] Proposez, complètement, de façon détaillée et justifiée, un code pour les temps.
- b) [2 points] Concrètement quel est votre code pour le temps 30 minutes et 58 secondes ?
- c) [5 points] Par rapport à la proposition du stagiaire faite en ①, combien gagnez vous de bits sur la représentation des temps de 1000 participants ? **Justifiez** votre réponse.

Réponses et justifications :

Ne pas écrire dans cette zone.



### Question 3 – C'est par où ? [11 points]

Vous envoyez un fichier par FTP depuis votre ordi (D, comme départ) à une amie (A, comme amie ou comme arrivée) via Internet.

Voici quelques **extraits** de tables de routage (aucune information pertinente n'est cachée ; toutes les informations nécessaires pour répondre à la question sont données) :

Table de D :

dest.	dir.	dist
A	X	3
W	Z	2

Table de X :

dest.	dir.	dist
D	D	1
A	Y	2
W	Z	2

Table de Y :

dest.	dir.	dist
D	X	2
A	A	1
W	W	1

Table de Z :

dest.	dir.	dist
D	D	1
A	W	2
W	W	1

Table de W :

dest.	dir.	dist
D	Z	2
A	A	1
X	Z	2

Table de A :

dest.	dir.	dist
D	W	3
X	Y	2

① [2 points] Quelle(s) route(s) suivent les paquets de votre fichier ?

Justifiez votre réponse.

Réponse et justification :

② [9 points] Sachant que :

- la liaison entre X et Y perd un paquet sur 10,
- la liaison entre Z et W perd un paquet sur 15,
- la transmission de votre fichier nécessite 150 paquets,
- et que les temps de traitement<sup>3</sup> d'un paquet par les nœuds sont les suivants (en ms) :

D	X	Y	Z	W	A
10	2	3	5	5	15

Combien de temps va prendre la transmission de votre fichier ?

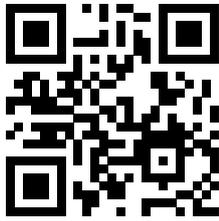
Justifiez pleinement votre réponse.

**Note :** on pourra faire l'approximation  $24/150 = 4/25 \simeq 1/5$

Réponse :

suite au dos ➡

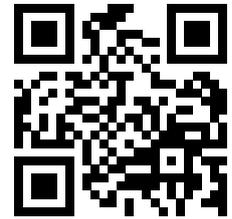
3. par « traitement », nous entendons, traitement *complet* par le nœud suivant ce qu'il a à faire (émission, réception, attente, retransmission, etc.)



---

(suite de la réponse si nécessaire :)

Ne pas écrire dans cette zone.



### Question 4 – Le retour de la machine [14 points]

On considère la machine de Turing ayant pour table de transition :

	0	1	$\varepsilon$
1	( 1, 0, +)	( 2, 0, -)	( 8, $\varepsilon$ , +)
2	( 2, 0, -)	( 1, $\varepsilon$ , -)	( 3, $\varepsilon$ , -)
3	( 4, 1, -)	( 3, 0, -)	( 4, 1, -)
4	( 4, 0, -)	( 4, 1, -)	( 5, $\varepsilon$ , +)
5	( 5, 0, +)	( 5, 1, +)	( 1, $\varepsilon$ , +)

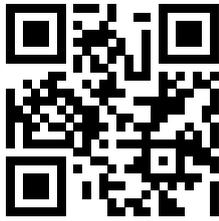
① [6 points] Quel est l'état de la bande et la position de la tête de lecture lorsque la machine s'arrête, si elle a démarré avec sa tête de lecture positionnée comme suit :

...	$\varepsilon$	0	$\varepsilon$	0	1	0	1	1	0	1	$\varepsilon$ ...
				↑							

② [8 points] Justifiez votre réponse en deux ou trois phrase(s), puis dites, en une courte phrase, ce que fait cette machine.

Réponses :

suite au dos ➡



### Question 5 – Le compte est bon [27 points]

On s'intéresse ici à écrire des *parties* d'un programme C++ permettant de résoudre le problème du « compte est bon » : à partir de 6 nombres entiers donnés et d'un résultat désiré (aussi entier), essayer de combiner un sous-ensemble quelconque de ces 6 nombres, pris chacun qu'une seule fois, pour obtenir le résultat.

Par exemple, peut-on trouver le résultat 329 à partir des nombres (2, 5, 5, 8, 25, 50) ?  
La réponse est « oui » :  $(5 + 2) \times (50 - (8 - 5)) = 329$ .

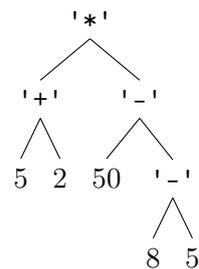
① [3 points] Dans un premier temps, on s'intéresse à représenter une suite de calculs arithmétiques. Pour cela, un *calcul* est défini comme :

- une opération (+, -, × ou /) (un `char` suffira en C++);
- un résultat (entier positif);
- un calcul, opérande de droite;
- un calcul, opérande de gauche.

Par exemple, le calcul  $(5 + 2) \times (50 - (8 - 5))$  peut se représenter comme :

- opération : '\*' ;
- résultat : 329 ;
- calcul de gauche : « (5 + 2) » ;
- calcul de droite : « (50 - (8 - 5)) ».

Évidemment, les deux derniers éléments sont également des *calculs*, chacun avec leurs éléments respectifs. Le calcul «  $(5 + 2) \times (50 - (8 - 5))$  » complet pourrait donc se représenter (sans les résultats) comme ci-contre :



Proposez un type `Calcul` pour représenter en C++ les calculs tels que définis ci-dessus.

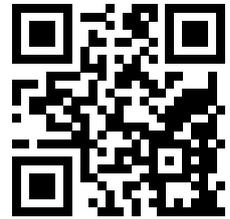
Réponse :

② [3 points] Comment représentez-vous les nombres de base utilisés dans les calculs (comme p.ex. 8, 5, 50 ou 2 dans le calcul précédent) ?

Définissez p.ex. en C++ la variable représentant le nombre 50 du calcul précédent.

Réponse :

Ne pas écrire dans cette zone.



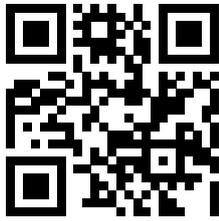
③ [7 points] On souhaite afficher joliment ces calculs. Par exemple, on souhaiterait afficher le calcul précédent  $((5 + 2) \times (50 - (8 - 5)) = 329)$  comme ci-contre, où le décalage de chaque nouveau sous-calcul est de 3 espaces vers la droite. Définissez la fonction C++ *recursive display()* permettant de faire un tel affichage.

Réponse :

```
329 =  
  7 =  
    5  
    +  
    2  
  *  
 47 =  
   50  
   -  
  3 =  
   8  
   -  
   5
```

suite au dos 

Ne pas écrire dans cette zone.



On ne va pas s'intéresser à tous les détails du programme complet, mais se focaliser uniquement sur la résolution « de haut niveau » dans les trois prochaines questions.

L'algorithme précis sera donné en question ⑤, mais le principe de résolution général est de réduire, par essais successifs, le problème de départ jusqu'à atteindre un calcul dont le résultat est le résultat désiré au départ. On procédera pour cela de proche en proche (récursivement) en remplaçant deux nombres du problème par le résultat de leur opération.

Par exemple, la résolution «  $(5 + 2) \times (50 - (8 - 5)) = 329$  » du problème de départ « peut-on trouver le résultat 329 à partir des nombres (2, 5, 5, 8, 25, 50) ? » aura été obtenue par résolution des problèmes suivants :

- « peut-on trouver 329 à partir de (2, 5, 5, 8, 25, 50) ? »
- « peut-on trouver 329 à partir de (7, 5, 8, 25, 50) ? » (combinaison de 5 et 2 en  $7 = 5 + 2$ )
- « peut-on trouver 329 à partir de (7, 3, 25, 50) ? » (combinaison de 8 et 5 en  $3 = 8 - 5$ )
- « peut-on trouver 329 à partir de (7, 47, 25) ? » (combinaison de 50 et 3 en  $47 = 50 - 3$ )
- « peut-on trouver 329 à partir de (329, 25) ? » (combinaison de 7 et 47 en  $329 = 7 \times 47$ )

dont la réponse est « oui ».

Il est clair que pour pouvoir reconstruire toute la solution, la liste donnée en seconde partie de chaque problème est plus qu'une liste de nombres : c'est une liste de `Calcul` (tels que définis en ①).

④ [1 point] Proposez un type C++ nommé `Objectif` contenant

- une liste de `Calculs` (ou, suivant votre choix, de pointeurs sur des `Calculs`), que nous appellerons « candidats » ;
- et un nombre entier, que nous appellerons « cible ».

Réponse :

Ne pas écrire dans cette zone.



⑤ [7 points] L'algorithme général de résolution est alors le suivant :

<b>solve</b>
entrée : <i>un Objectif</i>
sortie : <i>trouvé ou non ? (booléen)</i>
<pre> n ← nombre de candidats <b>Si</b> n ≤ 1     <b>Sortir</b> : Faux <b>Pour</b> i de 1 à n - 1     <b>Pour</b> j de i + 1 à n       <b>Pour</b> o parmi toutes les opérations (+, -, ×, /)         <b>Si</b> o est - ou /           <b>tester</b> l'opération o avec le candidat j et le candidat i           <b>Si</b> le test est positif             <b>Sortir</b> : Vrai         <b>tester</b> l'opération o avec le candidat i et le candidat j         <b>Si</b> le test est positif           <b>Sortir</b> : Vrai       <b>Sortir</b> : Faux     <b>Sortir</b> : Faux </pre>

où le sous-algorithme **tester** consiste à :

- vérifier que l'opération est valide (on ne peut par exemple pas effectuer de division dont le résultat ne soit pas un entier, ni de soustraction qui donne un résultat négatif)
- sortir** « Faux », sinon ;
- calculer le résultat de l'opération testée ;
- **si** le résultat dépasse (strictement) la cible, **sortir** « Faux » ;
- **si** le résultat égale la cible, **sortir** « Vrai » ;
- relancer la recherche (algorithme **solve** ci-dessus) avec comme nouvel objectif, la même cible, mais une liste de candidats réduite, où les deux candidats considérés ont été supprimés et le premier a été remplacé par le résultat du calcul de l'opération testée.

Quelle est la complexité de cet algorithme de résolution (**solve**) ?

**Justifiez** votre réponse.

**Réponse et justification :**

suite au dos



---

⑥ [6 points] En supposant qu'il existe une fonction

```
bool tester(char operation, const Calcul* first, const Calcul* second,  
           size_t i, size_t j, Objectif const& obj)
```

qui effectue l'algorithme **tester** sur l'opération **op** avec les deux candidats **first** et **second** (situés aux positions **i** et **j** des candidats de l'Objectif **obj**), écrivez le code C++ de la fonction **solve()** qui implémente l'algorithme **solve** décrit en ⑤.

Réponse :

Ne pas écrire dans cette zone.



---

**Question 6 – Variations algorithmiques [14 points]**

**Note :** Avant de répondre à ①, lisez la sous-question ③ (au dos).

On cherche à écrire un algorithme permettant de sortir, en ordre décroissant, les  $k$  valeurs maximales ( $k$  fixé, p.ex.  $k = 5$ ) de l'ensemble des produits des valeurs de deux listes.

Par exemple, pour les listes  $L_1 = (5, 2, 8, 3)$  et  $L_2 = (4, 7)$ , et pour  $k = 5$ , un tel algorithme sortira la liste  $(56, 35, 32, 21, 20)$ , puisque ce sont là les 5 valeurs maximales des produits des éléments de  $L_1$  par des éléments de  $L_2$ .

① [4 points] Écrivez un algorithme pour résoudre ce problème.

**Réponse :**

② [2 points] Quelle est la complexité de votre algorithme proposé en ①? Justifiez votre réponse.

**Réponse :**

suite au dos 



---

③ [8 points] Écrivez un algorithme de complexité temporelle  $\Theta(n \log n)$  pour résoudre le problème proposé, avec  $n$  la plus grande des tailles des deux listes.

Si votre réponse à ① est déjà en  $\Theta(n \log n)$ , vous n'avez rien à faire ici (et serez, bien entendu, noté(e) sur la somme des points des deux sous-questions).

**Réponse :**

Ne pas écrire dans cette zone.



## Question 7 – Taux de CO2 [17 points]

Une de vos amies veut créer un système de relevé automatique du taux de CO<sub>2</sub> dans sa chambre (entre 0 et 100). Elle sait que son capteur électronique a une précision de 0.1 et que chaque mesure sera stockée avec une indication du temps à laquelle elle aura été prise, exprimé en secondes depuis le 1<sup>er</sup> janvier 2000.

Le tout sera stocké en C++ dans un **vector** de **struct** ayant deux champs (dans cet ordre) :

- le taux comme un nombre à virgule flottante sur 32 bits avec 23 bits de mantisse (ordre : signe, exposant – ici directement en binaire non signé – puis mantisse)
- et le temps comme un **int** sur 32 bits.

① [3 points] Tout les combien de temps (au minimum) est-ce que son système doit sauvegarder une mesure si votre amie veut pouvoir reconstruire exactement la courbe des taux mesurés et que l'on suppose que ces mesures ne varient pas à plus de 0.025 Hz.

Justifiez votre réponse.

**Réponse et justification :**

② [7.5 points] Sachant que votre amie a commencé à enregistrer ses mesures à partir du temps écrit en binaire 0010 1000 0101 0000 0001 0010 1110 1011,

a) à quoi correspond l'enregistrement écrit en binaire sur le disque comme :

0000 0001 1010 0100 0000 0000 0000 0000, 0010 1000 0101 0000 0001 0101 0111 1111

Quel échantillon est-ce ? (c.-à-d. combien de temps après le début ?)

b) Quel est le taux de CO<sub>2</sub> correspondant ?

Justifiez vos réponses.

**Réponses et justification :**

suite au dos 



---

③ [6.5 points] Comment pourrait faire votre amie pour réduire la taille des informations stockées, sans perdre d'information ?

Essayez de lui proposer un système optimal. Expliquez votre démarche en détails.

**Réponse :**

Ne pas écrire dans cette zone.



---

Place supplémentaire pour répondre à n'importe quelle question si nécessaire. Mais  
**VEUILLEZ INDIQUER LE NUMÉRO DE LA QUESTION TRAITÉE.**

Ne pas écrire dans cette zone.



---

Place supplémentaire pour répondre à n'importe quelle question si nécessaire. Mais  
**VEUILLEZ INDIQUER LE NUMÉRO DE LA QUESTION TRAITÉE.**

Ne pas écrire dans cette zone.