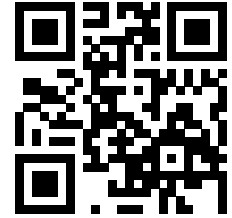


NOM : Hanon Ymous
(000000)
Place : 0

#0000



Information, Calcul et Communication (SMA/SPH) :

Examen I

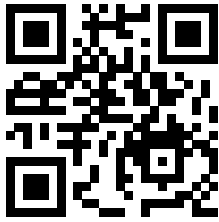
5 novembre 2021

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 (13h15 – 16h00).
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 8 exercices indépendants sur 12 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 150 points).
Tous les exercices comptent pour la note finale.



Question 1 – Coder [11 points]

Un étudiant a écrit l'algorithme suivant :

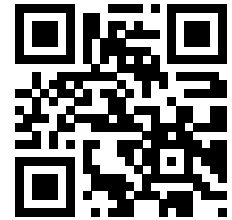
Machin
entrée : n , entier strictement positif
sortie : ???
$p \leftarrow 0$
$a \leftarrow 1$
$b \leftarrow 0$
Tant que $a < n$
$b \leftarrow b + 1$
Si b est un multiple de 3 ou de 5
$p \leftarrow p + b$
$a \leftarrow a + 1$
Sortir : p

① [4 points] Quelle valeur sort cet algorithme pour $n = 8$? Expliquez cette valeur en une phrase/formule.

② [7 points] Ecrire une fonction C++ qui implémente cet algorithme.

Réponses :

Réponses (suite) :



Question 2 – Ecriture d’algorithmes [45 points]

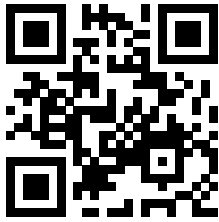
① [20 points] Ecrivez un algorithme qui effectue la fusion triée de deux listes triées.

Par exemple, si en entrée on a les listes $L_1 = (3, 7, 9, 12, 54)$ et $L_2 = (2, 9, 65)$, alors en sortie on aura la liste $(2, 3, 7, 9, 9, 12, 54, 65)$.

② [5 points] Quelle est la complexité de votre algorithme? Justifiez votre réponse.

Réponses :

suite au dos 

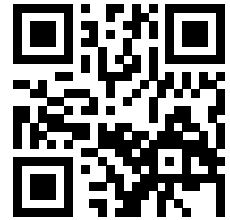


③ [20 points] Ecrivez un algorithme *récuratif* qui, prenant en entrée deux listes, effectue la fusion alternée de la première liste et du renversement (ordre de lecture inverse) de la seconde.

Par exemples :

- si en entrée on a les deux listes $L_1 = (1, 2, 3, 4)$ et $L_2 = (-11, -22, -33)$ (dans cet ordre), alors en sortie on aura la liste $(1, -33, 2, -22, 3, -11, 4)$;
- si en entrée on a les deux listes $L_1 = (12, 3, -5, 31, 6)$ et $L_2 = (8, 23, 11)$ (dans cet ordre), alors en sortie on aura la liste $(12, 11, 3, 23, -5, 8, 31, 6)$;
- par contre si en entrée on a les deux listes dans l'autre ordre : $L_1 = (8, 23, 11)$ et $L_2 = (12, 3, -5, 31, 6)$, alors en sortie on aura la liste $(8, 6, 23, 31, 11, -5, 3, 12)$.

Réponse :



Question 3 – Plan de quartier [10 points]

Une commune considère le problème « *boiser le quartier* » suivant :

elle veut offrir à chaque parcelle un arbre parmi 3 espèces (chêne, sapin ou hêtre) tout en s'assurant

- que deux parcelles voisines ne reçoivent pas un arbre de la même espèce ;
- et qu'aucun arbre ne soit à moins de 25 m d'un autre arbre d'une même espèce.

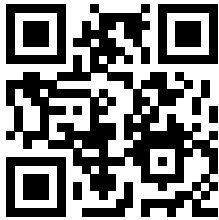
Pour résoudre ce problème, elle envisage l'algorithme suivant :

1. décider si l'on peut effectivement donner un arbre à chaque parcelle de sorte que deux parcelles voisines ne reçoivent pas un arbre de la même espèce ;
2. si c'est le cas, considérer une solution et calculer la distance minimale entre chaque paire de parcelles ayant reçu un arbre de la même espèce ;
3. vérifier que chacun de ces minima est supérieur à 25 m.

Que pouvez-vous dire de la complexité du problème « *boiser le quartier* » ?

Soyez le plus précis possible et **justifiez** votre réponse.

Réponse :



Question 4 – Complexité [20 points]

① [5 points] Pour trier un tableau de nombres de dimensions $n \times n$, on trie d'abord séparément chaque colonne avec un algorithme de tri optimal, puis ensuite chaque ligne avec ce même algorithme. Quelle est la complexité temporelle de cet algorithme de tri du tableau en entier ?

Justifiez votre réponse.

Réponse :

② Considérez le programme suivant (que l'on appliquera qu'à des entiers positifs ou nuls) :

```
int f(int i, int j)
{
    if (i <= j) return 1;
    int s(i + j);
    while (i > j) {
        s = s + 2;
        --i;
    }
    return s + f(j, s);
}
```

a. [3 points] Calculer $f(5, 3)$.

b. [12 points] Quelle est la complexité de l'algorithme implémenté par f ? **Justifier** votre réponse.

Réponses :



Question 5 – Petites machines [15 points]

① [6 points] On considère la machine de Turing dont la table de transition est :

	0	1	ε
1	(2, ε , +)	(3, ε , +)	(5, ε , +)
2	(2, 0, +)	(2, 1, +)	(4, 0, -)
3	(3, 0, +)	(3, 1, +)	(4, 1, -)
4	(4, 0, -)	(4, 1, -)	(5, ε , +)

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

...	ε	1	0	1	0	1	0	1	0	ε ...
		↑								

Expliquez votre réponse en une ou deux phrase(s).

Réponse :

② [9 points] On considère maintenant une autre machine de Turing, dont la table de transition est :

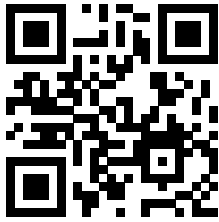
	0	1	ε
1	(1, 1, +)	(1, 0, +)	(2, ε , -)
2	(3, 1, -)	(2, 0, -)	(4, ε , +)
3	(3, 0, -)	(3, 1, -)	(4, ε , +)

Complétez le corps de la fonction C++ `f` ci-dessous, de sorte à ce que, pour un `int`, elle fasse la même chose que la machine de Turing ci-dessus (ça peut être simple ; en tout cas pas plus que la place laissée libre ici) :

```
int f(int a)
{

}

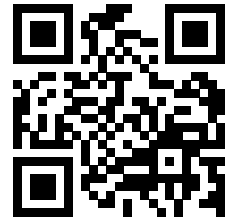
```



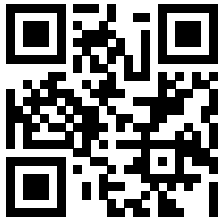
Question 6 – Représentations des nombres [19 points]

- ① [4 points] Sur une machine (moderne) où les `int` sont stockés sur 8 bits, quel est le schéma binaire en mémoire de la valeur de la variable `a` suivante :
`int a(-35);`
Justifiez brièvement votre réponse (p.ex. en explicitant vos calculs).
- ② [5 points] Combien vaut, en décimal, le résultat de l'opération en binaire signé par complément à 2, $01101000 + 01110101$?
Justifiez brièvement votre réponse.
- ③ [5 points] Le nombre 1.7 (en base 10) peut-il être représenté exactement (sans erreur d'arrondi) en binaire à virgule flottante avec 5 bits d'exposant et 10 bits de mantisse ?
Justifiez brièvement votre réponse.
- ④ [5 points] Quelle est la valeur décimale du nombre binaire représenté en virgule flottante sur 8 bits avec (dans cet ordre) 1 bit de signe, 2 bits d'exposant et 5 bits de mantisse, par `11001101` ?
Justifiez brièvement votre réponse (p.ex. en explicitant vos calculs).

Réponses :



(Place supplémentaire pour réponse si nécessaire.)



Question 7 Approximation – [15 points]

Le développement limité de la fonction « arc sinus » au voisinage de 0 est :

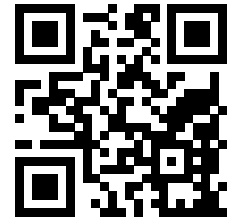
$$\arcsin(x) = x + \frac{1}{2 \cdot 3} x^3 + \frac{1 \cdot 3}{2 \cdot 4 \cdot 5} x^5 + \dots + \frac{1 \cdot 3 \cdot 5 \dots (2n-1)}{2 \cdot 4 \cdot 6 \dots (2n) \cdot (2n+1)} x^{2n+1} + o(x^{2n+2})$$

que l'on peut donc utiliser pour calculer une valeur approchée :

$$\begin{aligned} \arcsin(x) &\simeq x + \sum_{i=1}^N \frac{1 \cdot 3 \cdot 5 \dots (2i-1)}{2 \cdot 4 \cdot 6 \dots (2i)} \cdot \frac{1}{2i+1} \cdot x^{2i+1} \\ &= x + \sum_{i=1}^N \left(\frac{x}{2i+1} \prod_{j=1}^i \left(\frac{2j-1}{2j} x^2 \right) \right) \end{aligned}$$

Ecrivez une fonction C++ `asin_approx()` qui prend en argument x et N et utilise la formule ci-dessus pour calculer une valeur approchée de $\arcsin(x)$.

Réponse :



Question 8 – Trouver les erreurs [15 points]

On considère l'*extrait* de programme C++ suivant, qui comporte plusieurs erreurs de programmation de différente nature (syntaxe, déroulement, conception, méthodologie, ...):

```
1  double f(double x)
2  {
3      13 + 7*x - 2*x*x;
4  }
5
6  double derive(double x, double y, double epsilon)
7  {
8      return (y - x) / epsilon;
9  }
10
11 double df(double x)
12 {
13     const double epsilon(1e-5);
14     return derive(f(x), f(x+epsilon), epsilon);
15 }
16
17 double d2f(double x)
18 {
19     const double epsilon(1e-6);
20     return derive(df(x), df(x+epsilon), epsilon);
21 }
22
23 double extremum(double& a)
24 {
25     const double epsilon(1e-4);
26     const double u(0.0);
27     const double v(0.0);
28     int security(100.0);
29
30     do {
31         u = df(a);
32         v = d2f(a);
33         a -= u / v;
34         --security;
35     } while ((not (security = 1)) and (u/v == epsilon));
36
37     return a;
38 }
```

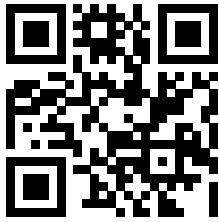
Ceci n'étant qu'un *extrait*, l'absence de `#include`, namespace et `main()` ne sont pas des erreurs.

Indiquez et corrigez toutes les erreurs (directement sur le code).

Expliquez brièvement les erreurs/correction à droite du code ou sur la page supplémentaire au dos.

On ôtera 1 point pour toute indication d'une erreur qui n'en est pas une.

En bonus, vous pouvez également indiquer des suggestions d'amélioration (qui ne sont pas, strictement parlant, des erreurs).



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