

Nom d'équipe : \_\_\_\_\_  
Université: \_\_\_\_\_

# INFORMATIQUE THÉORIQUE

CS GAMES 2025

Les Érudits de CS City ont besoin de votre aide! Ils ont déterminé que le secret de l'imagination se trouve dans votre cerveau et veulent vous étudier. Pour joindre l'utile à l'agréable, vous résoudrez des problèmes de CS City durant l'étude. Utilisez vos connaissances pour faire ressortir ce secret et améliorer la vie à CS City!

Instructions :

1. Les cellulaires, iPods, ordinateurs, montres intelligentes et autres appareils électroniques ne sont pas permis. Seule une calculatrice non programmable est acceptée.
2. Répondez directement sur le recto du questionnaire dans l'espace prévu à cet effet. Vous pouvez utiliser le verso comme espace supplémentaire si vous manquez de place. Assurez-vous de bien rendre visible la réponse aux questions.
3. Remettez ce questionnaire dans son entièreté dans l'ordre à la fin de l'épreuve.

Pour la correction			
<b>1. La logique ça se suit</b>	/20	<b>9. Automate en herbe</b>	/15
<b>2. Je connais ma machine</b>	/20	<b>10. Quelles sont les chances?</b>	/15
<b>3. Complexité d'algorithme</b>	/15	<b>11. Compilons tous ensemble</b>	/10
<b>4. Complexe construction</b>	/10	<b>12. Théo, l'homme décidable</b>	/4
<b>5. Turing the god machine</b>	/20	<b>13. La crypto c'est la vie</b>	/17
<b>6. Grand-mère vs Grammaire</b>	/15	<b>14. Théo, l'homme d'états</b>	/4
<b>7. Nettoyage en profondeur</b>	/20	<b>15. Petit puzzle funny</b>	/10
<b>8. Polyglotte 101</b>	/25	<b>16. Questions rapides</b>	/30
<b>TOTAL</b>			<b>/250</b>



## 1. La logique ça se suit

Travaillons d'abord votre logique pour voir si vous arrivez à imaginer quel sera le prochain nombre.

### 20 points (2 points par bonne réponse)

Trouvez le nombre qui continue les suites suivantes:

1. 4, 7, 11, 18, \_\_\_\_\_
2. 1, 11, 21, 1211, 111221, \_\_\_\_\_
3. 4, 16, 37, 58, 89, \_\_\_\_\_
4. 89, 106, 113, 118, 128, \_\_\_\_\_
5. 11, 17, 23, 31, \_\_\_\_\_
6. 9, 61, 52, 63, \_\_\_\_\_
7. 1, 2, 6, 42, 1806, \_\_\_\_\_
8. 0, 1, 10, 11, 20, 101, 21, \_\_\_\_\_
9. 4, 6, 15, 105, \_\_\_\_\_
10. 1, 2, 3, 4, 5, 6, 7, \_\_\_\_\_

## 2. Je connais ma machine

Pour bien analyser votre cerveau, les Érudits auraient d'abord besoin de votre aide pour faire fonctionner leur machine. Pourrez-vous la comprendre et leur expliquer?

### 20 points

1. Pouvez-vous expliquer à un enfant de 5 ans ce qu'est un langage assembleur. À quoi sert-il? (4 points)

2. Un des langages assembleur bien connu est l'ARM, voici donc un code ARMv7 simple (les numéros de ligne sont indiqués à gauche). Indiquez l'ordre des instructions exécutées par le microprocesseur en utilisant les numéros de ligne associés. (4 points)

1	MOV LR , PC
2	test
3	BX LR
4	BL test
5	CMP LR , PC
6	BLT test
7	MOV R0 , #0x1
8	B test

3. En vous basant sur le code suivant, répondez aux questions suivantes

```

1      B main
2
3      tableau DC32 0xFA , 0x32 , 0x05 , 0x45 , 0x02 , 0x00
4
5      main
6          LDR SP , =maPile
7          ADD SP , SP , #64
8
9          LDR R0 , =tableau
10         BL fonctionMystere
11         MOV R5 , R0
12
13         B main
14
15      fonctionMystere
16         PUSH {R1, R2, LR}
17         LDR R1 , [R0], #4
18         MOV R2 , R1
19
20      debut
21         CMP R1 , #0x00
22         BEQ fin
23
24         CMP R1 , R2
25         MOVLt R2 , R1
26         LDR R1 , [R0], #4
27         B debut
28
29      fin
30         MOV R0 , R2
31         POP {R1 , R2 , LR}
32         BX LR
33
34      maPile DS32 16

```

a. Pourquoi utilise-t-on les instructions PUSH et POP aux lignes 16 et 31? (2 points)

- b. Quelle est la valeur de R5 après l'exécution de l'instruction MOV R5, R0 à la ligne 11? (2 points)
- c. Décrivez ce que fait la fonction fonctionMystere. Attention de bien comprendre les arguments de la fonction et ce qu'elle retourne. (3 points)
4. Écrivez du code assembleur ARMv7 qui place dans R0 le signe (+ ou -1) de R1. En d'autres mots, implémentez le pseudo-code suivant: (5 points)

```
if R1 < 0 then
| R0 ← -1;
else
| R0 ← 1;
end
```

### 3. Complexité d'algorithme

Les Érudits aimeraient voir comment votre imagination vous aidera à comprendre un algorithme.

#### 15 points

1. Déterminez le temps d'exécution en fonction de  $n$  de la fonction suivante. Exprimez votre réponse en utilisant la notation asymptotique appropriée. Justifiez votre réponse. (10 points)

```
void f(unsigned int n)
{
    for ( unsigned int i=1; i<=(n*n); i = 2*i)
        for ( unsigned int j=1; j<=n; j = 2*j)
            cout << "bof" << endl;
}
```

2. Pour chaque situation décrite ci-dessous, indiquez quel conteneur de données il serait préférable d'utiliser parmi `<vector>`, `<list>`, `<stack>`, `<queue>`, et `<deque>`. Ce qui guide votre choix est le temps d'exécution des opérations que vous devrez effectuer. (1 point par bonne réponse)
- a. Après avoir inséré des éléments dans le conteneur, la seule opération qui sera faite sera d'accéder au k-ième élément stocké.
  - b. Vous utilisez le conteneur pour stocker des éléments temporairement et puis les récupérer dans l'ordre qu'ils ont été stockés.
  - c. Vous allez devoir stocker des éléments dans votre conteneur soit au début ou à la fin du conteneur. Vous allez également devoir accéder régulièrement au k-ième élément stocké.
  - d. Vous allez devoir régulièrement parcourir séquentiellement votre conteneur d'éléments et supprimer certains éléments satisfaisant une certaine propriété (qui n'est pas toujours la même). Vous devrez également insérer des éléments, mais la position d'insertion n'a pas d'importance.
  - e. Vous utilisez le conteneur pour stocker des éléments temporairement et puis les récupérer dans l'ordre inverse qu'ils ont été stockés.



## 4. Complexe construction

CS City demande votre aide pour réaliser une expansion de la ville. Bien sûr, avant de construire, il faut planifier comment les bâtiments seront construits!

### 10 points

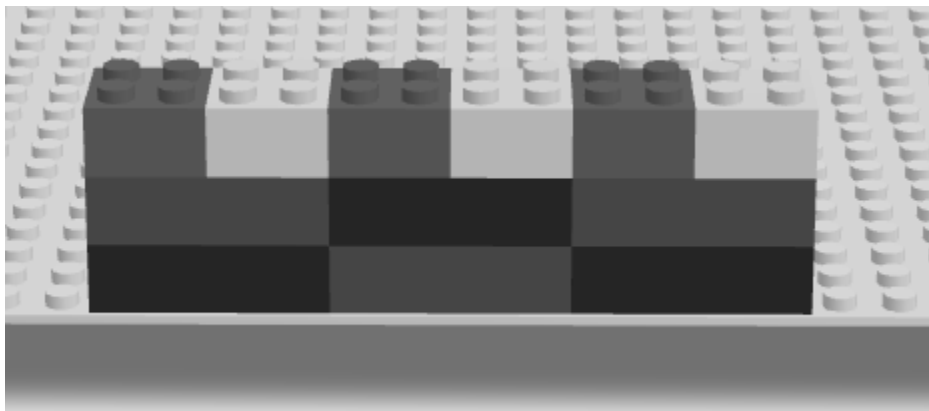
Vous devez créer un algorithme qui découvre et énumère toutes les façons de construire un mur de dimensions  $N \times M$ .

Les briques doivent être placées de manière que les craques de deux rangées consécutives ne soient pas alignées.

Pour construire ce mur, trois tailles de briques sont mises à votre disposition. Les tailles sont :  $\{1 \times 3; 1 \times 2; 1 \times 1\}$ . Vous avez une quantité infinie de briques de chaque taille.

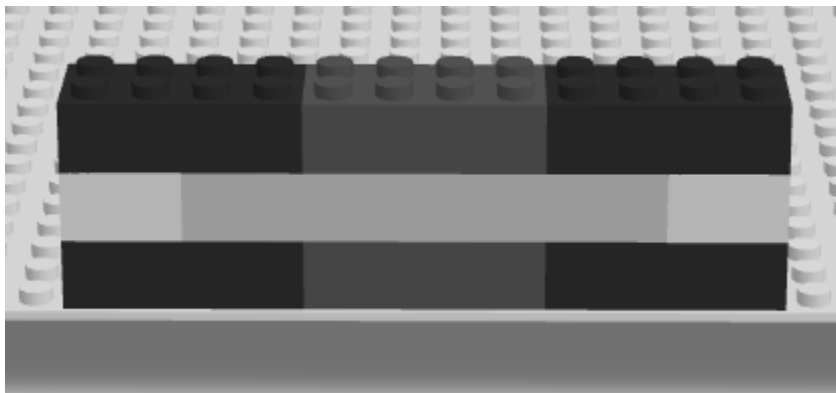
Vous devez donner la complexité algorithmique de votre solution et vous justifier.

#### Exemple 1:



Cet agencement n'est **PAS** valide, car les craques de deux rangées consécutives sont alignées.

#### Exemple 2:



Cet agencement est **valide**.

**NB: Les couleurs sont seulement utilisées pour faciliter la visualisation.**



## 5. Turing the god machine

Arriverez-vous à imaginer le fonctionnement d'automates? J'imagine que oui.

### 20 points

1. Notre nouvel ordinateur ne comprend pas comment valider les mots du langage suivant  $\{a^n b^n c^n : n \in \mathbb{N}\}$ , nous aurions besoin que vous construisiez un diagramme représentant une machine de Turing permettant de valider les mots de ce langage. (10 points)
2. Quelle est la différence entre un automate à pile et une machine de Turing? (Soyez précis sur les éléments de différences) (4 points)
3. Construire un automate à pile reconnaissant le langage  $L = \{a^i b^j \mid i = 2j\}$ . Vous préciserez bien le mode d'acceptation. Avant de donner la construction, vous expliquerez rapidement le fonctionnement de votre automate. (6 points)

## 6. Grand-mère vs Grammaire

Votre grand-mère aimerait avoir de l'aide pour construire sa grammaire. Votre grande mère n'a pas réussi à l'aider et les Érudits disent qu'ils galèrent.

### 15 points

1. On considère la grammaire G1 suivante

$$\begin{aligned} S &\rightarrow aAa \mid bBb \mid \varepsilon \\ A &\rightarrow C \mid a \\ B &\rightarrow C \mid b \\ C &\rightarrow CDE \mid \varepsilon \\ D &\rightarrow A \mid B \mid ab \end{aligned}$$

Mettre la grammaire G sous forme normale de Chomsky (4 points)

2. Construire une grammaire non contextuel G2 générant le langage (4 points)

$L = \{a^i b^j c^k \mid i = j + k, i, j, k \geq 0\}$  et donner une dérivation pour le mot aaabcc.

3. Considérez la grammaire  $G_3$  donnée par les règles ci-dessous. (L'alphabet terminal est  $\{a, b, \$\}$ .)

$$\begin{aligned} S &\rightarrow W T ba \\ S &\rightarrow abUV \\ T &\rightarrow W T ba \\ T &\rightarrow \$ \\ U &\rightarrow abUV \\ U &\rightarrow \$ \\ V &\rightarrow V a \\ V &\rightarrow ba \\ W &\rightarrow W b \\ W &\rightarrow ab \end{aligned}$$

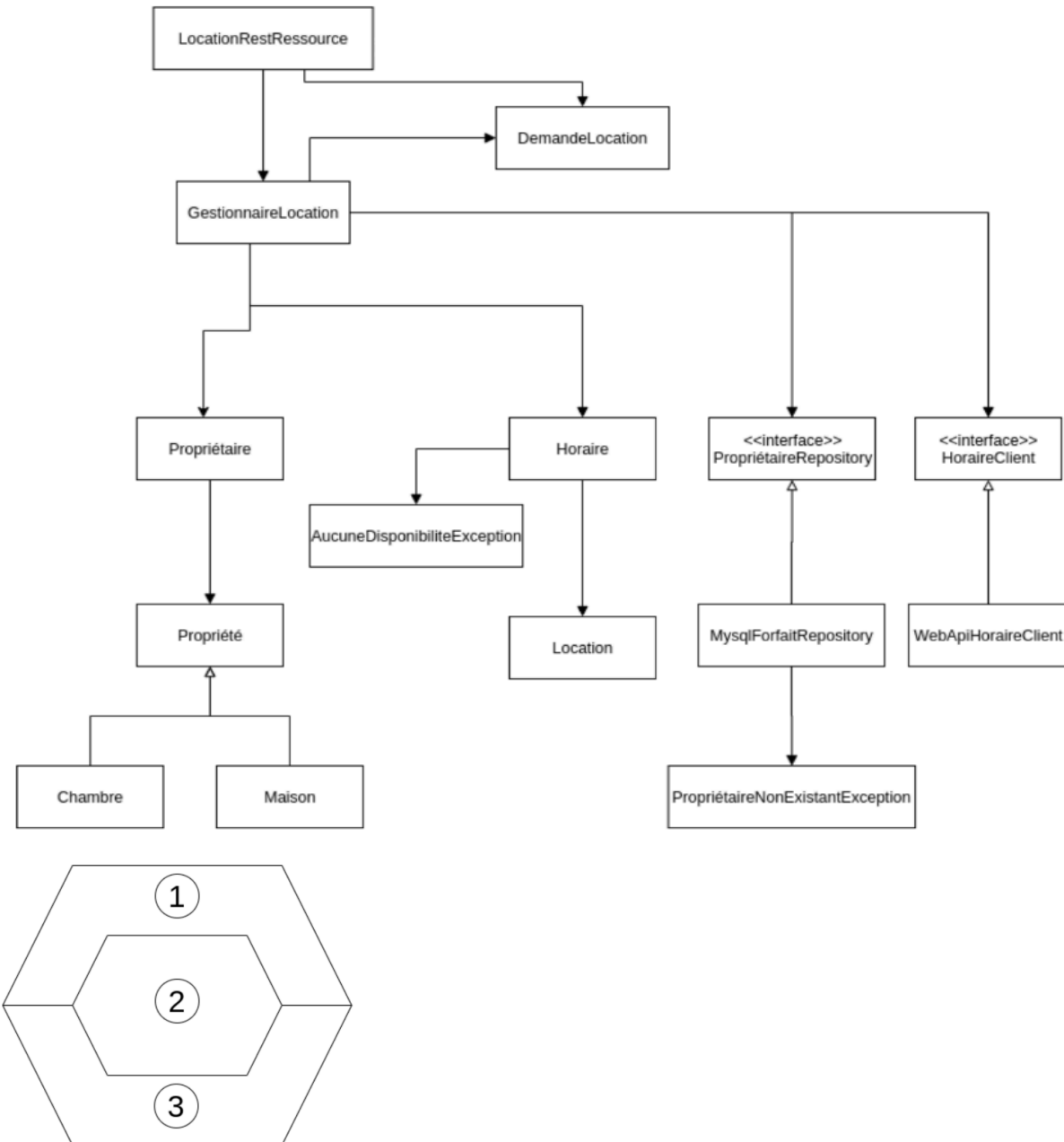
- a. Décrivez le langage généré par cette grammaire. (Par une expression ensembliste ou de toute autre façon qui est suffisamment claire et précise pour que le correcteur vous comprenne. Je suggère d'illustrer votre explication avec quelques exemples bien choisis de séquences qui appartiennent au langage et de séquences qui n'appartiennent pas au langage) (4 points)
- b. Montrez que  $G$  est ambiguë en donnant deux arbres de dérivation distincts pour un même mot. (3 points)

## 7. Nettoyage en profondeur

CS City est la ville la plus propre au monde, mais ce n'est pas sans effort! Avoir une ville propre, ça commence par nettoyer son logiciel!

### 20 points

1. En vous basant sur le diagramme suivant démontrant les différentes classes d'une application fictive, placer chacune d'entre elles dans une couche d'un modèle hexagonal. (8 points)



1: Interface 2: Domaine 3: Infrastructure

LocationRestRessource	1	2	3
DemandeLocation	1	2	3
GestionnaireLocation	1	2	3
Propriétaire	1	2	3
Propriété	1	2	3
Chambre	1	2	3
Maison	1	2	3
Horaire	1	2	3
AucuneDisponibiliteException	1	2	3
Location	1	2	3
PropriétaireRepository	1	2	3
MysqlForfaitRepository	1	2	3
PropriétaireNonExistantException	1	2	3
HoraireClient	1	2	3
WebApiHoraireClient	1	2	3

2. Si vous avez un Repository avec une interface comme celle-ci

```
public interface SuperRepository {
    User findByEmail(Email email);
    Order findByOrderNumber(OrderNumber orderNumber)
}
```

- a. En quoi cette interface ne respecte pas vraiment l'esprit d'un Repository? (3 points)
  
- b. Quel principe SOLID risque très fortement de ne pas être respecté pour cette raison? (2 points)

3. En considérant le code suivant:

```
class Panier:
    def emettreNouvelleFacture(self):
        articles_a_facturer = []
        for article in self.articles:
            if self.quantite[article] > 0:
                articles_a_facturer.append(article)
        facture = new Facture(self.client)
        facture.ajouter_articles(articles_a_facturer)
```

Quel(s) principe(s) SOLID n'est (sont) pas respecté(s)? (5 points)

Expliquez ce qui vous permet de l'affirmer en utilisant la définition du principe contextualisé à ce cas précis (ne donnez pas juste la définition).

Ne considérez que le code visible sans extrapoler aux autres classes.

4. Qui suis-je? (2 points)

En réalité, je suis composé de deux parties: une interface et des implémentations.

Le domaine connaît mon interface, mais jamais mon implémentation.

Mes méthodes sont orientées vers des besoins spécifiques au domaine d'affaires.

Je ne représente pas forcément une base de données, mais plutôt une source de données externe.

Mon objectif est de m'assurer que les décisions de cette source externe n'affectent jamais le domaine.

- Active Record
- DAO
- Repository



## 8. Polyglotte 101

Les Érudits sont impressionnés par votre imagination, mais ils aimeraient bien voir si cela a un impact sur votre capacité à écrire en utilisant différents langages de programmation.

### 15 points

Dans ce numéro, vous serez appelés à montrer que vous connaissez un grand nombre de langages et pour ce faire rien de mieux que d'écrire du code.

Dans chaque section, vous devrez écrire le code d'une fonction qui se nomme `reverseString` et qui prend un paramètre un `string` pour retourner un `string` qui a l'ensemble de ses caractères inversé (ex: Hello deviendrait olleH)

ATTENTION à ce que votre code soit clair et bien écrit, aucun point partiel. Si vous utilisez des librairies, n'oubliez pas de les importer.

(1 point par langage et 1 point bonus si les 14 sont réussis)

1. Python

2. Java

3. Rust

4. C++

5. Go

6. Ruby

7. Typescript

8. C

9. Elm

10. Fortran

11. Julia

12. R

13. Perl

14. VBScript

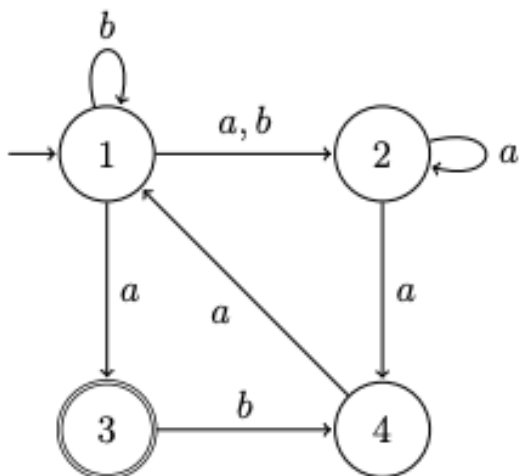
## 9. Automate en herbe

Bien que le titre du numéro l'indique, nous ne vous demandons pas de construire un automate constitué d'herbe (mais ce serait impressionnant).

### 15 points

1. Construisez des automates finis qui acceptent les langages suivants. Les automates peuvent être non déterministes et vous pouvez laisser tomber les états puits si cela vous simplifie la tâche. Par contre, vous ne pouvez pas utiliser des transitions étiquetées par une expression régulière ou par le mot vide  $\epsilon$ .
  - a. Sur l'alphabet  $\{0, 1\}$ , l'ensemble des mots non-vides dont la première lettre apparaît au moins deux autres fois. (4 points)
  - b. Soit  $L$  le langage sur l'alphabet  $\{0, 1\}$  défini par  $L = \{w \mid w \neq 00 \wedge w \neq 010\}$ . (Ce qui équivaut à dire que  $L$  comprend toutes les séquences sauf 00 et 010.) (5 points)

2. Construisez un automate déterministe qui accepte le même langage que l'automate nondéterministe ci-dessous. (6 points)



## 10. Quelles sont les chances?

Les habitants de CS City connaissent la chance de vivre dans une aussi belle ville, mais auraient besoin de votre sagesse pour comprendre la chance qu'ils ont.

### 15 points

1. Un bac de bloc contient  $n$  blocs rouges et  $n + 1$  blocs noirs. On sélectionne aléatoirement 4 blocs de l'urne sans remise. Quelle est la valeur de  $n$  si la probabilité d'obtenir 1 bloc rouge et 3 blocs noirs est de  $156/575$ ? (4 points)
  
2. Lors du dernier sommet du G8 qui eut lieu en Irlande du Nord en juin dernier, les places assises des 9 dirigeants présents (l'Union Européenne étant présente comme instance internationale) ont été attirées aléatoirement. Pour asseoir tout le monde, on disposait de deux tables rondes pouvant accueillir 6 personnes chacune (il y avait donc des chaises vides). Avant de faire le tirage, on a convenu que le représentant des États-Unis et celui du Canada devaient être assis côte à côte. Aucune autre contrainte n'a été émise. Les tables et les chaises sont toutes identiques, on ne s'intéresse donc qu'à la position relative des uns par rapport aux autres.
  - a. Combien y a-t-il de manières différentes d'asseoir les dirigeants ? (3 points)
  
  - b. Quelle est la probabilité que le représentant de l'Italie et celui de l'Allemagne soient assis à la même table sachant qu'une des deux tables accueille 6 personnes ? (4 points)



3. Serez-vous en mesure de prédire votre position dans le classement dans cette compétition après la correction de toutes les autres questions, excepté cette question. Nous avons besoin d'une position de 1er à 25e. (1 points)

4. Un héros blond vêtu de vert découvre un village montagneux. Afin d'obtenir la confiance des villageois, le chef lui remet cette mission:

“ Les Yooks, nos ennemis jurés, ont infiltré notre village en kidnappant un de nos amis et en assumant sa forme. Trouve l'espion, sachant que les Yooks sont tous des menteurs, alors que les villageois Anouki disent toujours la vérité. “

- *Chef des Anoukis*

Le héros interroge les villageois un par un et obtient les résultats suivants:

*Dobo*: “ Mazo est honnête. Ce type-là ne ment jamais! ”

*Kumu*: “ Mon intuition ne me trompe jamais. Mazo ou Aroo est le menteur que tu recherches! “

*Fofo*: “ Gumo et Zoma ne mentent pas, ce ne sont définitivement pas des Yooks!. “

*Zoma*: “ Je te promets que je ne mens jamais! “

*Mazo*: “ Crois-moi ou non, mais je te jure que Dobo et moi ne mentons jamais! ”

*Aroo*: “ Entre toi et moi, Kumu est un menteur... “

*Gumu*: “ Fofo, Mazo ou Aroo te ment constamment. “

Quel villageois est en fait le Yook déguisé? Spécifiez le raisonnement et les prédicats logiques qui vous ont permis d'arriver à cette réponse. (3 points)

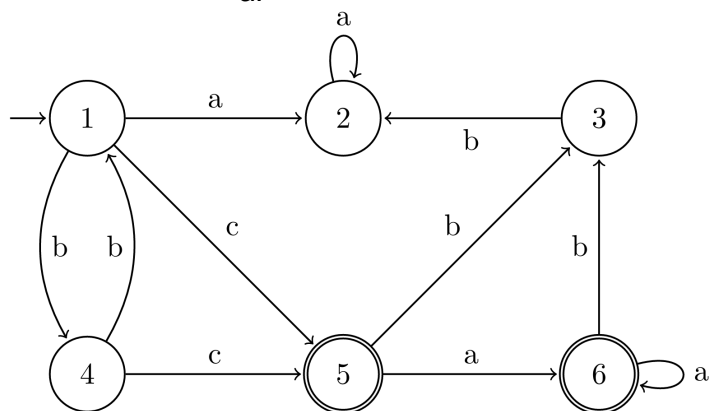
## 11. Compilons tous ensemble

Le dossier de devoirs des Érudits prend trop d'espace, ils auraient besoin de votre aide pour réduire sa taille.

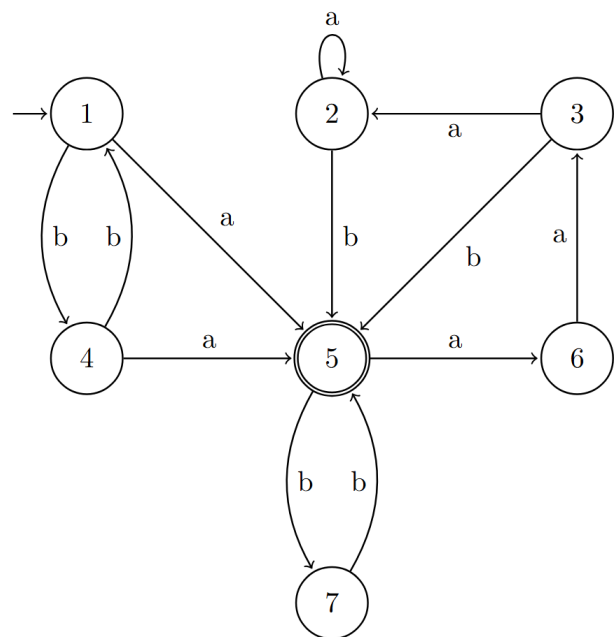
### 10 points

1. Minimisez les automates suivants en leur automate minimal (4 points chaque)

a.



b.



2. Donnez les expressions régulières correspondant aux langages des automates du numéro précédent (2 points)

## 12. Théo, l'homme décidable

Dans Divergente, Tris ne peut décider d'une faction puisqu'elle n'appartient à aucune. Théo, pour sa part, est décidable.

### 4 points

Montrez que le langage suivant est décidable :

$K = \{ \langle N \rangle : N \text{ est un automate fini déterministe fonctionnant sur l'alphabet } \{0, 1, 2, 3, 4\} \text{ et } L(N) \text{ contient au moins un sous-mot (une séquence continue de symboles) palindromique d'au moins 3 caractères.} \}$ .

### 13. La crypto c'est la vie

Grâce à la science, on a encrypté de l'imagination dans ces messages. Personne n'est certain à quoi ça sert, mais les Érudits aimeraient bien que vous la récupériez.

#### 17 points

1. Voici plusieurs mots encodés de manières différentes, retrouvez-les

- a. 506f72747567616c (1 points)
- b. Noitceles (1 points)
- c. Ubpxrl (1 points)
- d. 078101116102108105120 (1 points)
- e. PfGrqdog (1 points)

2. À l'aide du chiffre de Vigenère et de la clé "GAUSS":
  - a. Chiffrez le message en clair "DISQUISITIONESARITHMETICAE" (2 points)
  
  
  
  
  
  
  
  
  
  
  - b. Déchiffrez le message chiffré "VAOUS YEX ESZULS" (2 points)
  
  
  
  
  
  
  
  
  
  
3. À l'aide du chiffre du XOR et de la clé "11001100"
  - a. Chiffrez le message binaire "10110101" (1 points)
  
  
  
  
  
  
  
  
  
  
  - b. Déchiffrez le message chiffré "1011110101011101" (1 points)
  
  
  
  
  
  
  
  
  
  
4. Donnez le 2025e nombres pseudo-aléatoires générés par les générateurs congruentiels linéaires suivants et expliquez l'importance d'un mauvais choix de paramètres sur ce type de générateurs (4 points)
$$X_n = (3X_{n-1} + 7)\%11 \text{ avec } X_0 = 5$$

5. On associe à la majorité des articles des magasins un code universel des produits, dont la représentation sous forme de code à barres doit vous être familière:



Le code précédent est dans le système UPC-A. Dans ce système, le dernier chiffre est une somme de contrôle que l'on calcule de la façon suivante. Le rang des chiffres débute à gauche: 1er = 7, 2eme = 8 etc.

On additionne ensuite les chiffres de rang impair et on multiplie la somme par 3. On ajoute ensuite à cette somme la somme des chiffres de rang pair. Si le résultat est congru à 0 modulo 10, alors le numéro est valide donc 780672318862 est valide.

Est-ce que le code UPC-A suivant est valide? (1 points)



6. Écrivez une Regex qui correspond à toutes les chaînes sauf les chaînes qui commencent par LEGO. La Regex doit être construite sans lookahead (?= et ?!) ou lookbehind (?<= et ?<!). (2 points)

## 14. Théo, l'homme d'états

Depuis que vous avez démontré qu'il est décidable, Théo est dans tous ses états!

### 4 points

Pour ce numéro, l'alphabet est  $\{0,1\}$ . Soit  $x$  un mot non vide sur cet alphabet, on définit  $n_x$  comme la représentation binaire du mot  $x$ . Démontrer qu'il est possible de représenter ce langage par un automate fini déterministe.

$$K = \{x : x \neq \epsilon \text{ \&\& } (n_x) \bmod 42 = 5\}$$

Bien sûr, vous n'avez pas l'obligation de dresser la liste de toutes les transitions possibles (malgré qu'il s'agisse d'une réponse valide), vous pouvez expliquer comment les générer.

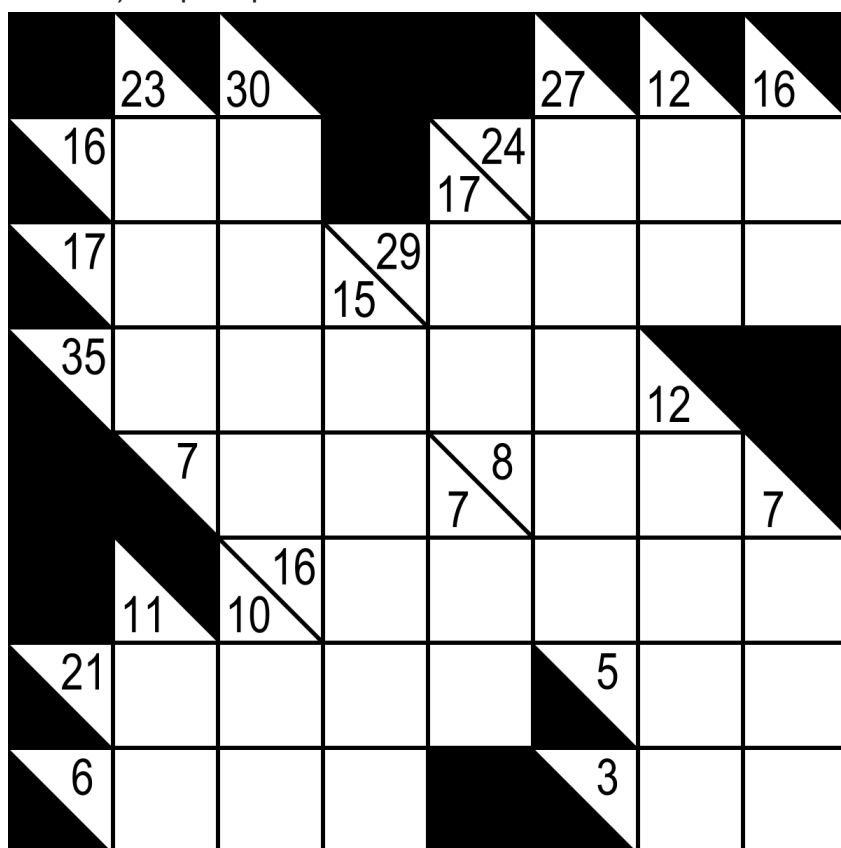
## 15. Petits puzzles funny

Les Érudits aimeraient vraiment voir si vous pouvez résoudre ces casse-têtes. Pour la science.

### 10 points

#### 1. Kakuro (4 points)

L'objectif est de remplir les cases vides (blanches) avec des chiffres entre 1 et 9 de sorte que la somme de tous les chiffres d'une ligne (suite horizontale et contigüe de case vide) soit égale au nombre inscrit dans la case à l'extrémité gauche de celle-ci et que la somme de tous les chiffres d'une colonne (suite verticale et contigüe de case vide) soit égale au nombre inscrit dans le haut de celle-ci. Une addition (ligne ou colonne) ne peut pas contenir deux fois le même chiffre.





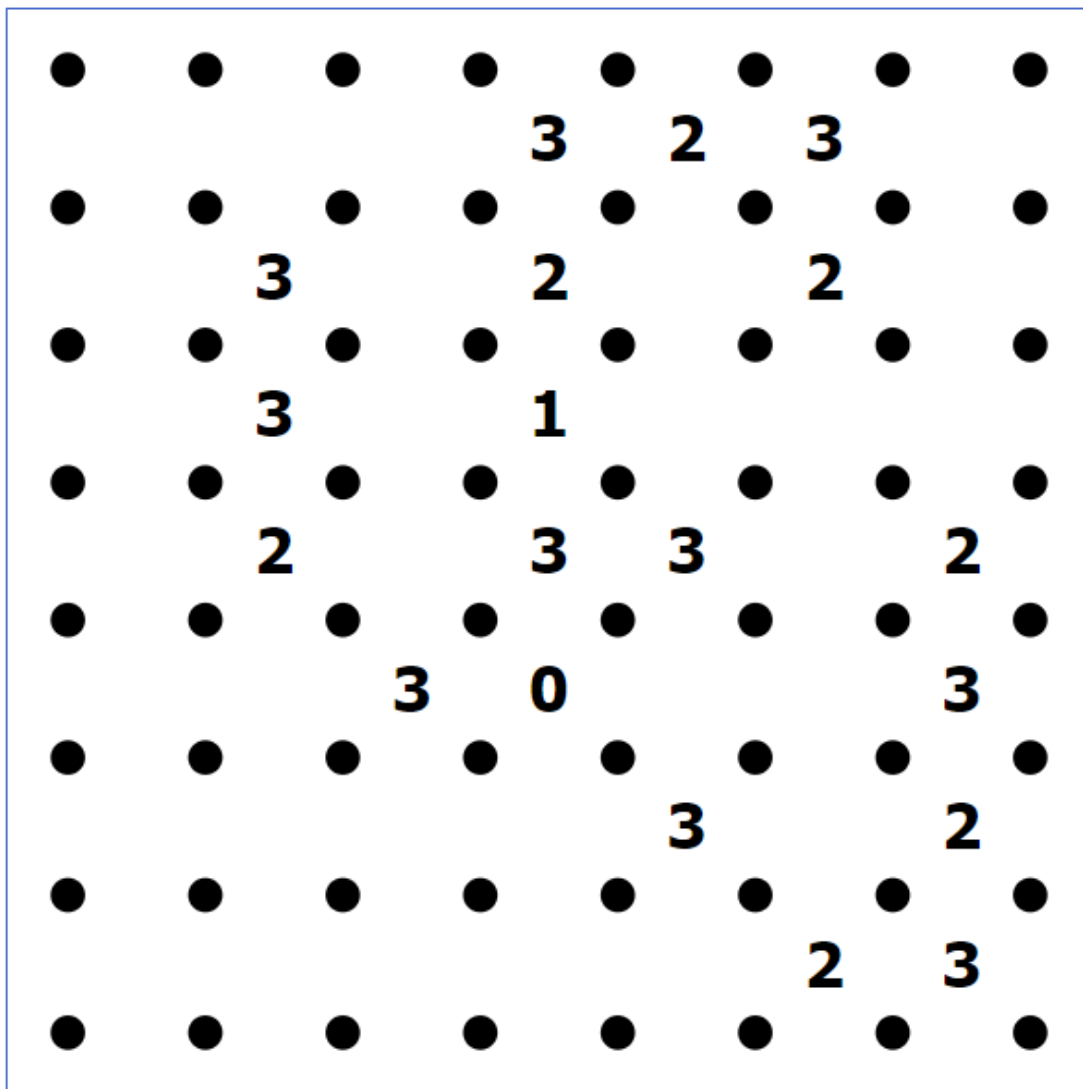
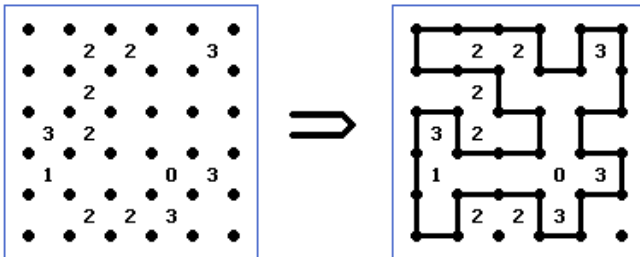
## 2. Sudoku (3 points)

Le but du jeu est de remplir la grille. Chaque ligne, chaque colonne, et chacun des neuf sous-grilles de 3x3 délimité par des lignes en gras, doit contenir tous les chiffres de 1 à 9 sans répétition.

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

## 3. Puzzle loop (3 points)

Il faut créer une ligne continue qui ne se croise jamais et qui forme une boucle en respectant le nombre de lignes passant sur le contour des carrés avec des chiffres.



## 16. Questions rapides

Comme les habitants de CS City ne mentent jamais, les Érudits sont fascinés par votre habileté à distinguer le vrai du faux. Serait-ce un produit votre imagination?

### 30 points (1 points par bonne réponse)

Vrai ou Faux

1. Une pile est une structure de données de type *FIFO*, “first in, first out”.
  - a. Vrai
  - b. Faux
2. Le langage assembleur facilite l'écriture d'un programme pour un microprocesseur donné par rapport au langage machine.
  - a. Vrai
  - b. Faux
3. Il est impossible pour  $f(n)$  d'être en  $\Theta(n \log n)$  si  $f(n)$  est en  $\Omega(n)$  et en  $O(n^{3/2})$ .
  - a. Vrai
  - b. Faux
4. En supposant que  $f(n)$  dénote le temps d'exécution d'un algorithme lorsque l'instance à traiter est de taille  $n$ . Si  $f(n)$  est en  $\Theta(n \ln(n))$ , alors  $f(n)$  est en  $\Omega(n)$ .
  - a. Vrai
  - b. Faux
5. En supposant que  $f(n)$  dénote le temps d'exécution d'un algorithme lorsque l'instance à traiter est de taille  $n$ . Il est impossible pour  $f(n)$  d'être en  $\Omega(n \log n)$  si  $f(n)$  est en  $O(n)$ .
  - a. Vrai
  - b. Faux
6. Un Méga-octet contient 1.048.576 octets.
  - a. Vrai
  - b. Faux
7. Redis est une technologie de cache distribuée.
  - a. Vrai
  - b. Faux

8. Dans le contexte des bases de données, une transaction **ACID** garantit que toutes les mises à jour sont visibles pour tous les utilisateurs du système en même temps.
  - a. Vrai
  - b. Faux
9. Un arbre binaire de recherche (BST) est toujours équilibré.
  - a. Vrai
  - b. Faux
10. En C++, dans le cas usuel, la fonction `std::move` transfère la propriété d'une ressource, mais ne déplace pas réellement les données en mémoire.
  - a. Vrai
  - b. Faux

#### Code d'erreur

11. Que signifie le code d'erreur HTTP 404
  - a. *Unauthorized*
  - b. *Forbidden*
  - c. *Bad Request*
  - d. *Not Found*
12. Que signifie le code d'erreur HTTP 200
  - a. *OK*
  - b. *Forbidden*
  - c. *Accepted*
  - d. *Created*
13. Que signifie le code d'erreur HTTP 303
  - a. *Moved Permanently*
  - b. *Multiple Choices*
  - c. *See Other*
  - d. *Not Modified*
14. Que signifie le code d'erreur HTTP 451
  - a. *Precondition Required*
  - b. *Blocked by Windows Parental Controls*
  - c. *Retry With*
  - d. *Unavailable For Legal Reasons*
15. Que signifie le code d'erreur HTTP 501
  - a. *Unauthorized*
  - b. *Not Implemented*
  - c. *Bad Gateway*
  - d. *Service Unavailable*

## Connaissance des langages

16. Qui est connu comme le créateur du langage Java?

- a. Thomas Sowell
- b. James Gosling
- c. Bill Joy
- d. Emmet Brickovski

17. Qui est connu comme le créateur de Python?

- a. Guido van Rossum
- b. Linus Torvald
- c. Dennis Ritchie
- d. Tim Wozniak

18. Qui est connu comme le créateur du Ruby?

- a. Jack Druggens
- b. Simon Peytons
- c. Yukihiro Matsumoto
- d. David Flanagan

19. Que veut dire PHP?

- a. elePHant Program
- b. Process for Hyper-Programming
- c. PHP: Hypertext Pre-processor
- d. PHP (les lettres n'ont pas de signification)

20. *Durant quelle décennie la disposition de clavier QWERTY a-t-elle été inventée?*

- a. 1870-1879
- b. 1910-1919
- c. 1950-1959
- d. 1980-1989

21. Quelles innovations ont mené à la gestion de la mémoire dans les systèmes d'exploitation modernes?

- a. La pagination automatique et la segmentation de la mémoire
- b. La mémoire paginée, la mémoire virtuelle et la segmentation de la mémoire
- c. La mémoire virtuelle et la DDR3
- d. La micro-segmentation, la gestion dynamique et la mémoire cache.

## Autres

22. Dans un algorithme de tri rapide (QuickSort), quel est le rôle du pivot ?

- a. Le pivot est un élément sélectionné au hasard, qui est placé à sa position finale.
- b. Le pivot permet de diviser le tableau en sous-tableaux plus petits pour un tri récursif.
- c. Le pivot trie lui-même les sous-tableaux.
- d. Le pivot est utilisé pour créer un tableau de valeurs moyennes, facilitant le tri.

23. Quel est l'impact de l'utilisation d'une indexation non valide dans un tableau brut dans un langage comme C++ ?

- a. Le programme génère une exception d'exécution.
- b. Cela entraîne un comportement indéfini et peut provoquer des erreurs de segmentation.
- c. Le programme renvoie une valeur par défaut comme `null` ou `0`.
- d. L'index est ignoré et la mémoire est allouée dynamiquement.

24. Quel algorithme est utilisé pour le chiffrement asymétrique ?

- AES
- RSA
- SHA-256
- MD5

On vous donne la réponse

$$25. \neg(\neg a \wedge \neg b) = a \vee \neg b$$

- Vrai
- Faux

26.  $(\neg\neg\neg\neg\neg\neg\neg\neg\neg b) \vee (a \wedge b)$  si  $a$  est Vrai et  $b$  est Faux

- a. Vrai  
b. Faux

27.  $(b \vee (a \wedge b)) \oplus ((\neg b \vee a) \wedge b)$  si  
a est Faux et b est Vrai

- Vrai
- Faux

## Utilitaires

## ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[END OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

## Math

$$P(n, r) = \frac{n!}{(n-r)!} \quad P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$C(n, r) = \frac{P(n, r)}{r!} = \frac{n!}{r!(n-r)!}$$

$$(a+x)^n = a^n + na^{n-1}x + \frac{n(n-1)}{2!}a^{n-2}x^2 + \frac{n(n-1)(n-2)}{3!}a^{n-3}x^3 + \dots$$

$$= a^n + \binom{n}{1}a^{n-1}x + \binom{n}{2}a^{n-2}x^2 + \binom{n}{3}a^{n-3}x^3 + \dots$$

$$\sum_{k=1}^n k = \frac{1}{2}n(n+1)$$

$$\sum_{k=1}^n k^2 = \frac{1}{6}n(n+1)(2n+1)$$

$$\sum_{k=1}^n k^3 = \frac{1}{4}n^2(n+1)^2$$

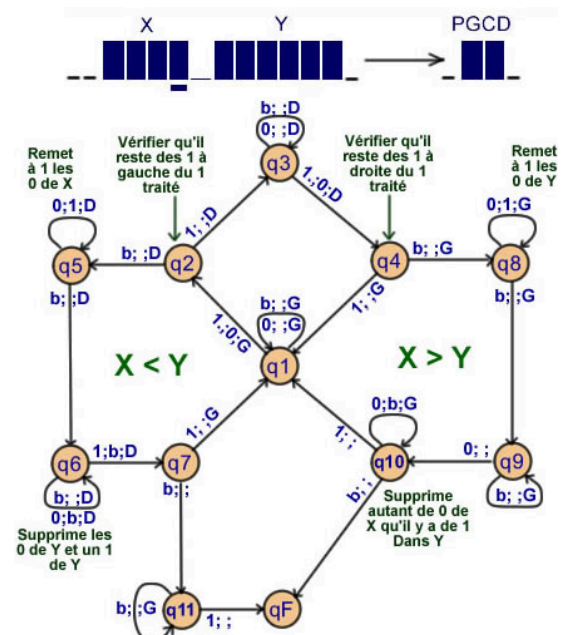
Table ARMv7

Instruction	Description
ADD Rd, Rs	$Rd \leftarrow Rs + Rd$
ADD Rd, Const	$Rd \leftarrow Rd + Const$
ADD Rd, Rs, Op1	$Rd \leftarrow Rs + Op1$
AND Rd, Rs, Op1	$Rd \leftarrow Rs \text{ AND } Op1$
B etiquette	$PC \leftarrow adresse(etiquette)$
BL etiquette	$LR \leftarrow PC - 4, PC \leftarrow adresse(etiquette)$
BX Rs	$PC \leftarrow Rs$
CMP Rs, Op1	drapeaux = $Rs - Op1$
LDR Rd, etiquette	$Rd \leftarrow valeur(etiquette)$
LDR Rd, =etiquette	$Rd \leftarrow adresse(etiquette)$
LDR Rd, [Rb, Op1]	$Rd \leftarrow Mem[Rb + Op1]$
LDR Rd, [Rb], Op1	$Rd \leftarrow Mem[Rb], Rb \leftarrow Rb + Op1$
LDR Rd, [Rb, Op1]!	$Rb \leftarrow Rb + Op1, Rd \leftarrow Mem[Rb]$
MUL Rd, Rn, Rs	$Rd \leftarrow Rb \times Rs$
MVN Rd, Op1	$Rd \leftarrow !Op1$ (inverse the bits)
POP {Liste Reg}	$SP \leftarrow SP - 4 \times (\text{nb registre})$
PUSH {Liste Reg}	$SP \leftarrow SP + 4 \times (\text{nb registre})$
STR Rs, etiquette	$valeur(etiquette) \leftarrow Rd$
STR Rs, [Rb, Op1]	$Mem[Rb + Op1] \leftarrow Rs$
STR Rs, [Rb], Op1	$Mem[Rb] \leftarrow Rs, Rb \leftarrow Rb + Op1$
STR Rs, [Rb, Op1]!	$Rb \leftarrow Rb + Op1, Mem[Rb] \leftarrow Rs$
SUB Rd, Rs	$Rd \leftarrow Rd - Rs$
SUB Rd, Const	$Rd \leftarrow Rd - Const$
SUB Rd, Rs, Op1	$Rd \leftarrow Rs - Op1$
MOV Rd, Rs	Write register value (Rs) in register value (Rd)
MOV Rd, Const	Write constant in Rd
Condition	Code Condition Code
Equals	EQ Not equals NE
Greater than	GT Less than LT
Greater or equals	GE Less or equals LE

## Informatique théorique

Une grammaire non contextuelle est en **forme normale de Chomsky** si et seulement si toutes ses règles de production sont de la forme :  $X \rightarrow YZ$  ou  $X \rightarrow a$  ou  $S \rightarrow \epsilon$  où  $X, Y, Z$  sont des symboles non terminaux,  $a$  est un symbole terminal,  $S$  est l'axiome de la grammaire et  $\epsilon$  est le mot vide.

## Exemple Machine de Turing





## Pages supplémentaires

(N'oubliez pas d'écrire la question que vous finissez ici)





