Juin 2024

Les automates au lycée

Objectif : Aborder le principe général de fonctionnement d'un automate puis modéliser un automate.

Capacités travaillées : Lire, comprendre, modifier, compléter et concevoir un algorithme.

Compétences : Savoir utiliser une règle de codage pour traduire une information.

Mettre en oeuvre un raisonnement, articuler les différentes étapes d'une solution.

Formuler et communiquer sa démarche et ses résultats par écrit et les exposer oralement.

Niveaux : lycée (tous niveaux).

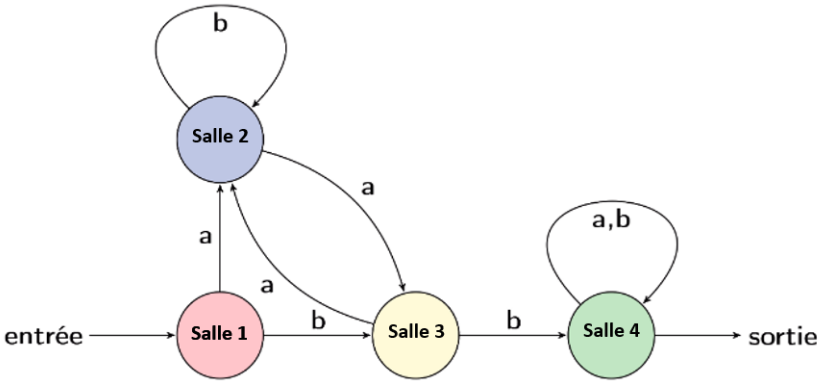
Ressources pour l'enseignant : Fiche scientifique pour avoir quelques éléments théoriques sur les automates finis. https://sancy.iut.uca.fr/~iso/docs/automates/FicheScientifique.pdf ;

Article sur *Tangente Éducation* n°42-43. Informatique débranchée. Pages 46 à 51.

Partie 1 : Qu'est-ce qu'un automate ? Une première activité de découverte.

Les automates sont utilisés dans de nombreux dispositifs de notre environnement quotidien : digicodes, distributeurs, parkings… Dans cette partie 1, les élèves se familiarisent avec les diagrammes qui modélisent les différents états d'un automate grâce à un problème de labyrinthe.

L'enseignant dessine au tableau ou distribue le plan du petit labyrinthe ci-dessous composé de salles et de passages possibles d'une salle à une autre. Ces passages sont représentés par une flèche indiquant un sens de parcours et portant une étiquette (a ou b). Dans cet exemple, partant d'une salle, il y a toujours deux possibilités a ou b.



L'enseignant demande aux élèves de proposer des suites gagnantes formées des lettres a et b de plus en plus longues qui permettent de sortir de ce labyrinthe, il les note au tableau au fur et à mesure. (bb, aab, abab, …)

Puis il propose lui-même des suites (gagnantes et perdantes) et demande aux élèves de déterminer si ces dernières sont gagnantes ou perdantes.

Exemples de suites gagnantes : **bb**ababa ; **bb**aaaa ; **ba**baaab ; **a**bbbaaaaabab ; …

Exemples de suites perdantes : **ba**baab ; **ba**aaaab **; a**bbaaaab ; **a**bbbbaaaaaaaab ; …

L'objectif est ici que les élèves en viennent à énoncer collectivement une règle de construction des suites gagnantes formée des lettres a et b aussi complète et précise que possible.

Par exemple : « on peut commencer par **bb** et ensuite faire n'importe quoi, ou bien commencer par **a** ou par **ba** et ensuite faire autant de b et de aa qu'on veut, puis après faire un ab et après on termine par tout ce qu'on veut. »

L'enseignant insiste sur le fait qu'on arrive à trier toutes les suites de lettres dans deux groupes disjoints : celles qui permettent de sortir du labyrinthe et les autres.

Bilan de cette activité : Nous venons « d'imiter » le comportement de machines ou de programmes qui ont pour rôle de **reconnaître automatiquement des suites de lettres (ou de chiffres) particulières afin de réaliser une tâche.**

En s'appuyant sur le travail qui vient d'être effectué, l'enseignant peut alors donner une définition d'un automate aux élèves qui serait :

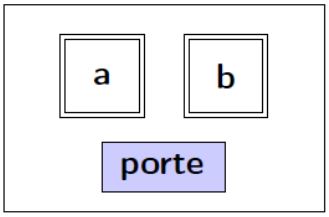
« Un automate est un algorithme représenté par un graphe orienté et étiqueté (ou un schéma codé comprenant des états sous forme de cercles et des transitions sous forme de flèches étiquetées) permettant de répondre à une question par oui ou par non. »

Cette question serait dans l'exemple traité précédemment : « Est-ce que la suite de lettre suivante babaaab permet de sortir du labyrinthe ? » (oui !)

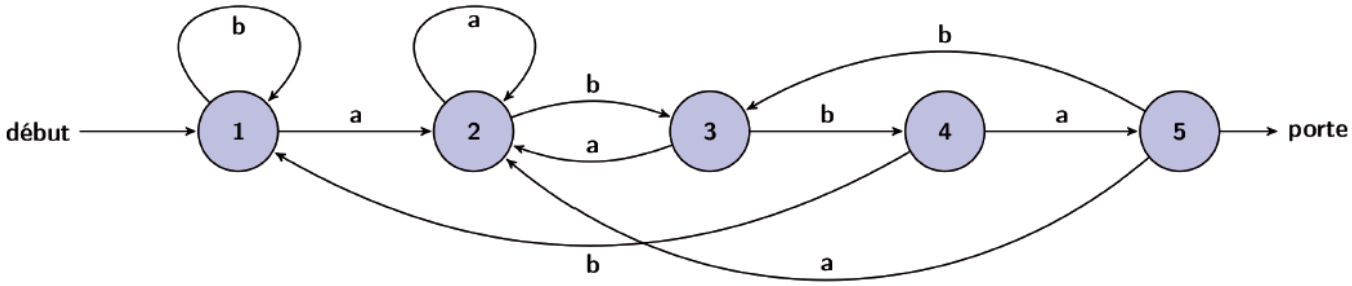
Partie 2 : Analyser et concevoir un automate simple.

On trouve des automates dans le fonctionnement d'objets de la vie de tous les jours (un digicode, un distributeur de boissons ou de billets), et dans des fonctions très courantes des ordinateurs comme chercher un mot dans un texte. L'enseignant illustrera son propos, dans cette partie 2, en dessinant au tableau le schéma du digicode (simplifié) à deux touches et l'automate (le labyrinthe) associé.

On imagine une sorte de digicode à deux touches (marquées a et b), qui débloque le bouton d'ouverture d'une porte lorsqu'on frappe une suite de lettres convenable, appelé code.



L'automate ci-dessous représente le comportement de ce digicode associé à une famille de codes donnés.



Analyser un automate simple :

L'enseignant peut alors demander aux élèves :

« Donner un code qui permet de débloquer le bouton d'ouverture de la porte. Est-ce le seul ?» abba convient, ce n'est pas le seul ;

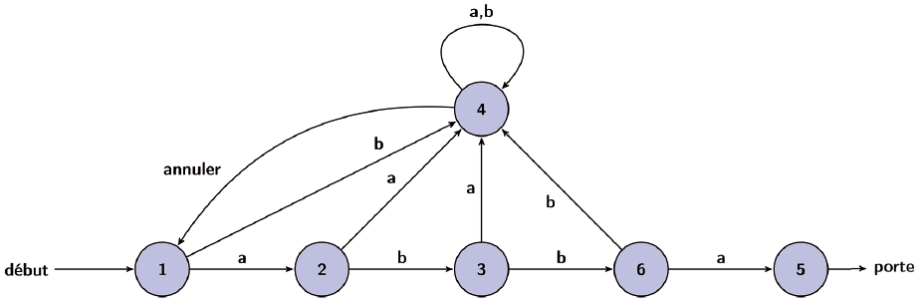
« Que se passe-t-il si on se trompe ? » Rien de particulier, la porte s'ouvre sitôt qu'on tape une suite de lettres qui se termine par abba.

Ainsi, les suites aaabba ou abababba ou évidemment abba vont convenir, mais pas les suites abb ou aabbaba.

Par conséquent, on peut dire que, parmi toutes les suites de lettres a et b possibles, ce digicode a pour rôle de reconnaître celles qui se terminent par abba.

Remarque : Les élèves demandent quelquefois si, en ouvrant le boitier, on verrait le schéma. Réponse : Non pas directement, on verrait seulement un dispositif électronique ou mécanique.

Une autre question concerne les digicodes qui « se bloquent » si on ne tape pas le bon code, jusqu'à ce qu'on appuie sur le bouton « annuler ». Ils existent effectivement et on peut proposer d'en chercher collectivement un automate (voir ci-dessous).



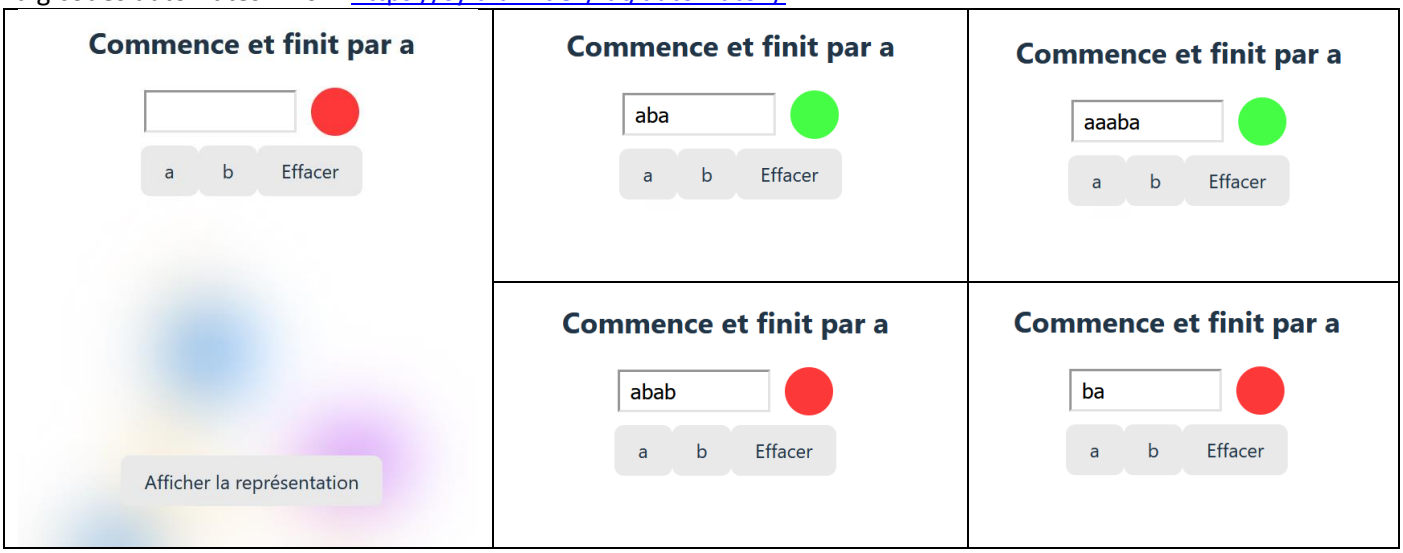
Concevoir un automate simple :

Il s'agit dans cette partie d'aborder la construction d'un automate sur un exemple simple.

Pour cela, on décortique la description en français d'un ensemble de codes pour en déduire l'automate représentant le comportement d'un digicode qui reconnaît ces codes et seulement ceux-là.

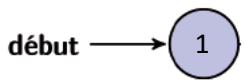
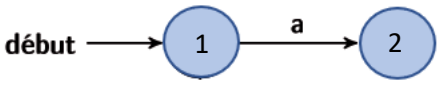
L'enseignant choisit un digicode qui ouvre la porte pour tous les codes qui commencent et finissent par un a et seulement pour ceux- là.

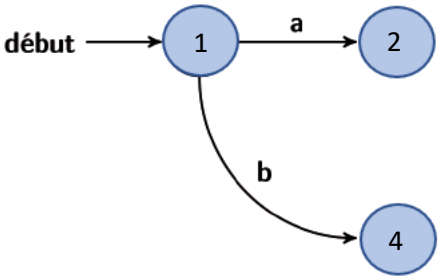
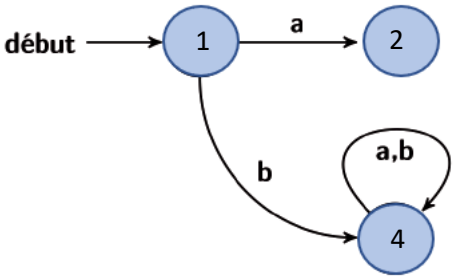
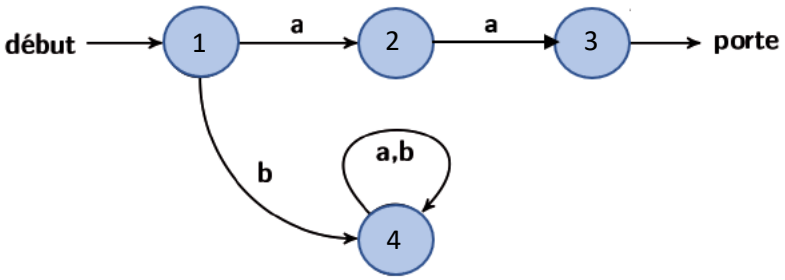
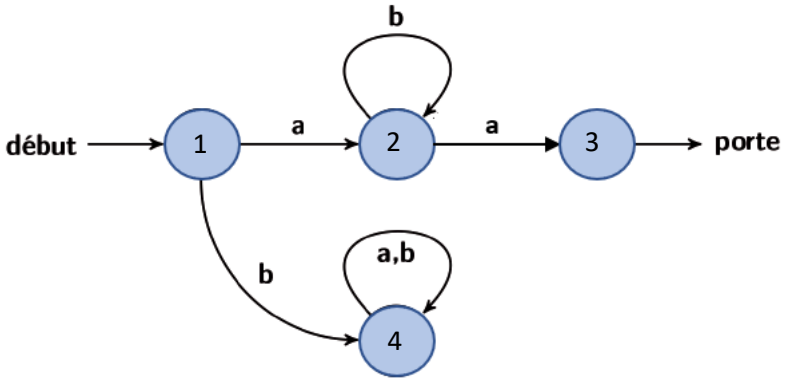
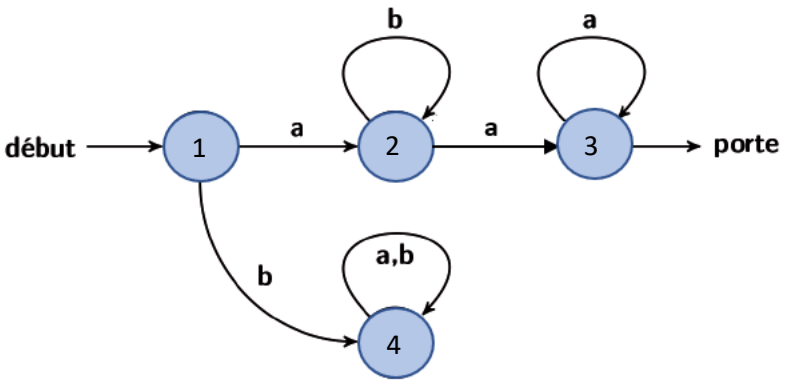
Pour tester ce digicode l'enseignant peut utiliser (sans faire afficher l'automate dans un premier temps) l'application "digicodes automates finis" : https://syldium.dev/iut/automaton/

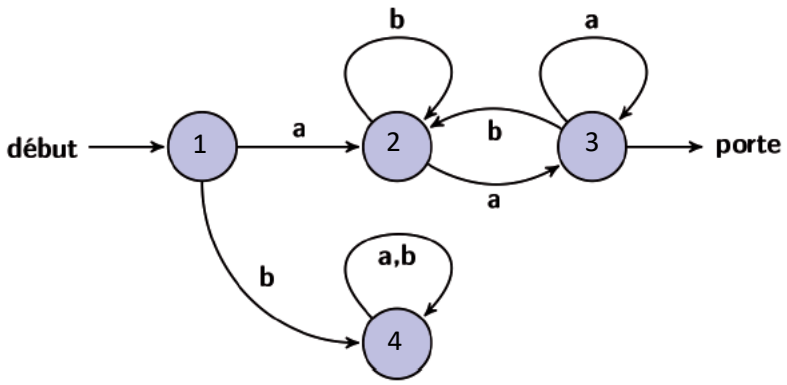


Il s'agit de construire maintenant avec les élèves l'automate associé à cette situation.

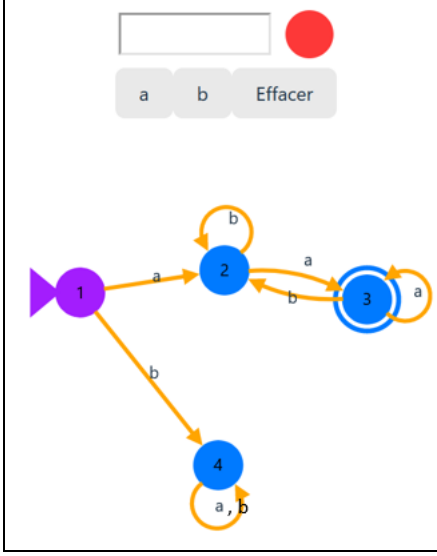
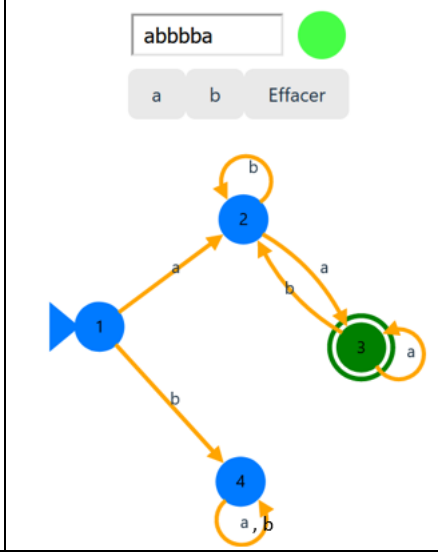
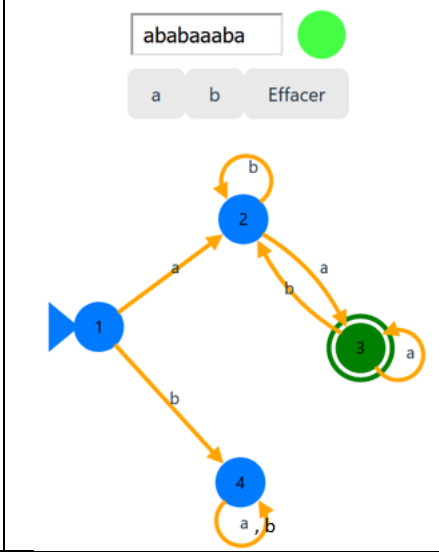
Le tableau ci-dessous décrit les étapes de construction, pas à pas, avec les élèves.

On commence par dessiner l'état initial (état 1).	
Si la première lettre est un **a**, tout va bien et on passe dans l'état suivant (état 2).	

Mais si la première lettre est un **b**, il est certain que le code n'est pas bon, et on va à l'état « **poubelle »** (état 4).	
Quand on est dans l'état « **poubelle »**, on ne peut jamais en sortir, donc les transitions étiquetées **a** et **b** bouclent dans cet état.	
Quand on est dans l'état 2, si on tape un autre **a** et qu'on s'arrête là, le code **aa** est bon, donc on arrive dans l'état terminal (état 3).	
Quand on tape un **b** à partir de l'état **2**, il ne se passe rien de spécial, on reste dans le même état.	
Quand on est dans l'état terminal, si on continue à taper des lettres **a**, on reste dans cet état.	

Par contre, si on tape un **b**, on retourne dans l'état 1, en attendant le prochain **a**.	

On peut utiliser revenir à l'application "digicodes automates finis" en faisant apparaître l'automate et en visualiser les étapes lorsqu'on rentre un code : https://syldium.dev/iut/automaton/

Le code commence et finit par a	**Le code commence et finit par a**	**Le code commence et finit par a**
		

Entrainement, approfondissement :

Les élèves sont ensuite invités à faire un travail d'informaticien : **dessiner des automates pour des digicodes qui ouvriront la porte pour des ensembles de codes définis à l'avance.**

Ce travail peut se faire en classe en groupe ou bien à la maison. Il est possible de proposer le même type de code à toute la classe, ou bien de poser une question différente à chaque groupe.

Questions possibles : « **Dessiner l'automate pour un digicode qui ouvrira la porte pour l'ensemble des codes définis par : - le code commence par bb ;**

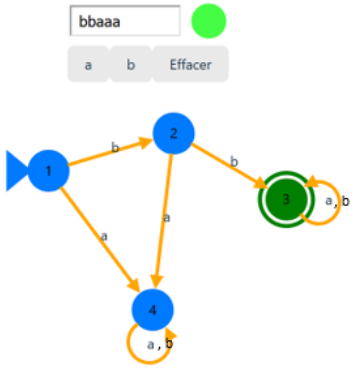
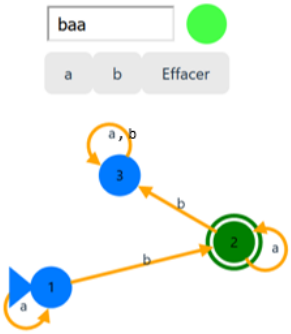
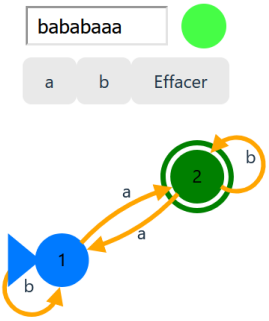
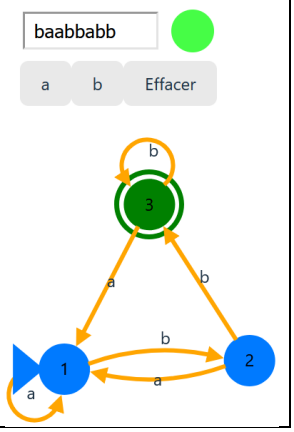
- le code a exactement un b ;

- le code a un nombre pair de a ;

- le code finit par deux b. »

Conseil que l'enseignant peut donner aux élèves : commencer par faire une liste de quelques suites de lettres permettant d'ouvrir la porte et de quelques-unes ne le permettant pas, pour amorcer la réflexion et vérifier que l'automate proposé soit correct. L'enseignant vérifie avec les élèves que la forme des codes attendue est bien comprise.

Les élèves peuvent manipuler et valider leur travail en utilisant l'application "digicodes automates finis" : https://syldium.dev/iut/automaton/

Le code commence par bb	Le code a exactement un b	Le code a un nombre pair de a	Le code finit par deux b
			

Partie 3 : Application – Quelques exemples concrets :

Cette dernière partie peut faire l'objet de devoirs maisons donnés tout au long de l'année.

Le vérificateur d'horaire :

Un des domaines où la modélisation à l'aide d'automates finis est très utilisée est ce qu'on appelle l'algorithmique du texte au sens large : recherche documentaire, traitement de texte, génomique, compilation, etc.

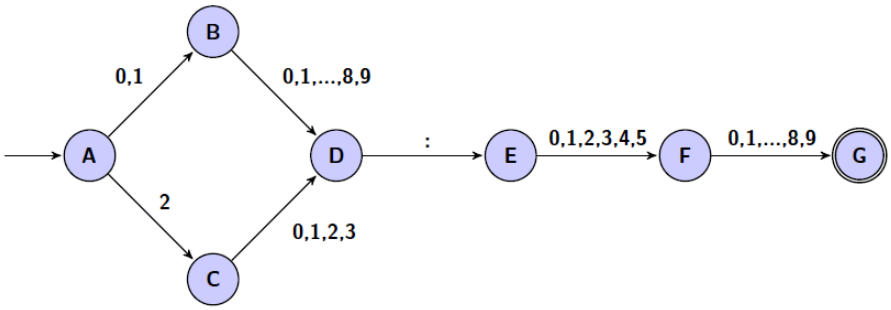
Activité : On s'intéresse à un formulaire sur un site internet qui demande d'entrer un horaire sous la forme HH:MM et vérifie si la valeur donnée est correcte avant de donner accès à la touche Entrée.

Les horaires acceptables vont de 00:00 à 23:59, par conséquent des expressions comme 15:44 ou 08:00 sont correctes, mais pas 5:12, ni 42:05, ni 06:75, ni 04:255.

Le comportement du programme de vérification de la valeur donnée par l'internaute peut être modélisé par un automate.

Construire cet automate. Ici, les noms des états sont des lettres, pour ne pas créer de confusion avec les étiquettes dont certaines sont des chiffres.

Solution :



Quelques remarques de construction :

Il faut impérativement deux chiffres, suivis du symbole « : », puis encore deux chiffres.

Le premier chiffre (celui des dizaines des heures) ne peut prendre que les valeurs 0, 1 ou 2 :

- Si c'est un 0 ou un 1, alors le chiffre des unités correspondant peut prendre n'importe quelle valeur entre 0 et 9 ;
- Si c'est un 2, alors les seuls chiffres des unités possibles sont 0, 1, 2 et 3.

Ensuite, le symbole « : » est obligatoire.

Pour les minutes, les seuls chiffres des dizaines possibles vont de 0 à 5, tandis que le chiffre des unités est quelconque.

L'enseignant, partant de ce premier exemple simplifié, peut proposer aux élèves des modèles plus réalistes à analyser, en se reportant à la fin de la fiche scientifique. **https://sancy.iut.uca.fr/~iso/docs/automates/FicheScientifique.pdf.**

La partie de tennis :

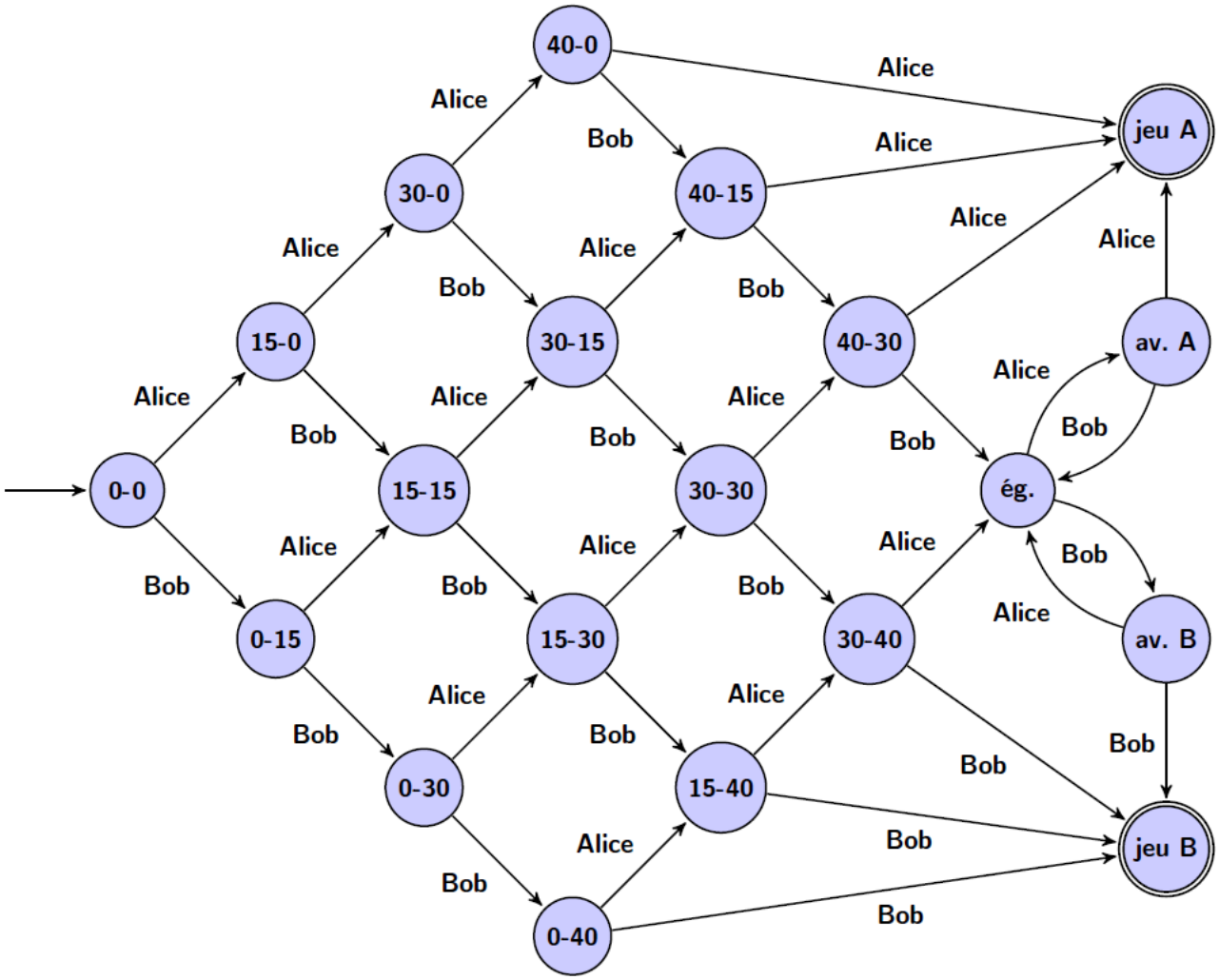
Un automate fini peut aussi permettre de modéliser un système concernant des processus complètement abstraits, comme les différentes étapes du scénario d'un jeu.

Activité : Construire un automate modélisant l'évolution du score au cours d'un jeu pendant une partie de tennis entre Alice et Bob.

Les états correspondent aux différents scores possibles : comme « 40-15 », ou « av. A » pour avantage Alice, ou « eg. » pour égalité, ou « jeu A » pour jeu Alice ;

Les évènements Alice ou Bob représentent un point marqué par le joueur Alice ou Bob respectivement.

Solution :



Remarques possibles aux élèves en lien avec l'orientation professionnelle :

Concrètement, lors de la conception d'un nouveau jeu (par exemple un futur jeu sur console, mais pas seulement), on peut modéliser à l'aide d'automates de ce type les différents scénarios envisagés, ou certaines parties de ces scénarios.

D'une façon plus générale, les diagrammes d'utilisation que produisent les concepteurs de logiciels pour analyser les futures interactions avec les usagers, ou bien entre les différents modules d'un logiciel, sont aussi très souvent modélisables par des automates finis.

La machine à café :

De nombreuses machines mécaniques ou électroniques peuvent être représentées par des automates. La plupart des automates obtenus dans la vie réelle sont évidemment beaucoup plus compliqués que ceux qu'il est possible de présenter ici.

Activité : Construire l'automate associé à la machine à café simplifiée suivante :

Elle n'accepte que les pièces de 10 centimes et 20 centimes. Un café coûte 30 centimes.

Quand le montant est suffisant, elle ferme la fente permettant d'introduire des pièces, débloque un bouton permettant d'obtenir le café, rend éventuellement la monnaie et retourne dans l'état initial si on appuie sur le bouton café.

Si on glisse une pièce d'un autre type que 10 ou 20 centimes, la machine passe dans un état intermédiaire, dont on ne peut sortir qu'en appuyant sur la touche « annuler », qui déclenche le rendu des pièces et le retour à l'état initial.

Solution :