

25 mai 2022



Rapport de stage

LYCEE XXXXX, XXXXX
CHOURRÉ MAXIME

Sous la tutelle de M. XXXXX Xxxxx, enseignant

Stage dans l'enseignement, en classe de SNT et NSI

Comment faciliter la compréhension du nouveau programme d'informatique par un public de lycéens ?

Remerciements

Je remercie le lycée XXXXXXXX, sous la direction de XXXXXXXX Xxxxxxx, de m'avoir accueilli du 3 décembre 2019 au 13 mars 2020.

Je remercie également Monsieur XXXX Xxxxxx d'avoir été mon tuteur pendant le stage.

Sommaire

I.	Introduction du stage.....	1
1.	La plateforme d'accueil : Le lycée XXXXXXXX	1
2.	La problématique	1
3.	Les conditions de stage	1
II.	Analyse des programmes	3
1.	Sciences numériques et technologie (SNT).....	3
a.	Notions transversales de programmation	3
b.	Informatique embarquée et objets connectés	3
c.	Internet.....	4
2.	Numérique et sciences informatiques (NSI)	4
a.	Démarche de projet.....	5
b.	Représentation des données.....	5
c.	Langages et programmation.....	5
d.	Algorithmique.....	6
III.	Les cours et exercices.....	6
1.	Sciences numériques et technologie (SNT).....	6
a.	Arduino	6
b.	Réseau.....	8
2.	Numérique et sciences informatiques (NSI)	8
a.	Arduino	8
b.	Réseau.....	9
c.	Python.....	9
IV.	Conclusion du stage	11
	Bibliographie	12
	Annexe A : Organigramme.....	13
	Annexe B : Tableau des notions de programmation.....	14
	Annexe C : Tableau des attentes sur l'informatique embarquée	14
	Annexe D : Tableau des attentes sur internet.....	15

Annexe E : Représentation des données	16
Annexe F : Langages et programmation.....	17
Annexe G : Extrait activité réseaux.....	18
Annexe H : Le logiciel Filius	19
Annexe I : Cours Arduino - Les bases.....	20
Annexe J : Cours Arduino - Le LCD	20
Annexe K : Détecteur de distance	21
Annexe L : Tinkercad.....	22
Annexe M : Evaluation Marionnet	23
Annexe N : Extraits de Bases de programmation.....	25
Annexe O : Extrait du cours sur les Listes	26
Annexe P : Evaluation Python	27

I. Introduction du stage

C'est après l'annonce de la réforme des lycées, et la recommandation de XXXXXXX que j'ai décidé d'organiser mon stage de licence au lycée XXXXXXX, à XXXX. La règle était simple : un stage de 30 h minimum, afin d'accompagner les enseignants au lycée, dont l'informatique n'est pas leur matière principale.

1. La plateforme d'accueil : Le lycée XXXXXXX

XXXXXX

2. La problématique

Le 16 juillet 2018 a été actée la réforme du baccalauréat qui vise à mettre en place un système à spécialités. Suite à cela, deux nouvelles matières d'informatique ont vu le jour :

- SNT¹ est un enseignement obligatoire aux élèves en classe de seconde, à raison de 1 h 30 par semaine ;
- NSI² est un enseignement facultatif disponible en classe de première (4 h par semaine) et en classe terminale (6 h par semaine).

Suite à cette réforme, les enseignants ont dû s'adapter à de nouvelles matières qu'ils ne connaissaient pas ou peu. J'ai donc décidé de réaliser un stage avec l'un de ces enseignants, afin d'observer l'autre côté de la classe, mais également afin d'aider ce dernier à **faire apprendre aux élèves**.

3. Les conditions de stage

Le stage s'est déroulé en deux parties. Dans un premier temps, une période d'observation les 3 et 20 décembre 2019 où j'ai pu observer le fonctionnement des cours, et la méthodologie du professeur. Dans un second temps, une période de pratique, qui est l'objet principal de ce rapport. Les dates étaient entre les vacances

¹ Sciences numériques et technologie

² Numérique et sciences informatiques

de Noël, et les vacances de Pâques. 2 h chaque jeudi (de 10 h à 12 h) avec une classe de NSI, et 2 h chaque vendredi (de 8 h à 10 h) avec deux classes de SNT.

Il est important de noter qu'en raison du coronavirus³, je n'ai pu finir mon stage. C'est-à-dire que les séances initialement prévues les 19, 20, 26, 27 mars et 2 et 3 avril n'ont pu être réalisées (12 h). La durée totale de stage réalisée s'élève à 29h30 devant la classe.

³ Arrêté du 14 mars 2020, article 4

II. Analyse des programmes

1. Sciences numériques et technologie (SNT)

Le programme complet est disponible dans la bibliographie (Ministère de l'Éducation nationale, 2019). Les notions importantes seront placées en annexe.

Dans le préambule du programme, on remarque que celui-ci a pour but de « appréhender les principaux concepts des sciences numériques » et de montrer « le poids croissant du numérique et les enjeux qui en découlent ». Cela signifie que c'est une matière qui vise à informer les élèves à propos des nouvelles technologies, et à les aider à les manipuler.

Cet enseignement vise également à expliquer les « enjeux scientifiques et sociétaux de la science informatique ». On peut donc la considérer comme une matière de culture générale, mais malheureusement ce n'en est pas le cas.

En effet, on trouve « la pratique de la programmation » qui sera intégrée à travers les différentes notions du programme. C'est donc une matière (pour rappel obligatoire) qui vise à apprendre la programmation aux élèves de secondes.

Personnellement, je trouve que rendre obligatoire la programmation n'est pas utile pour les élèves souhaitant avoir un programme orienté lettres ou social.

Pendant mon stage, j'ai pu travailler sur les thématiques d'internet et de l'Informatique embarquée et objets connectés.

a. Notions transversales de programmation

Les notions transversales de programmation définissent les notions que les élèves devront apprendre à propos du langage, qu'est ici Python 3. On y retrouve les bases de la programmation (variables, instructions, boucles, fonction) dans le but de « écrire et développer des programmes » (Annexe B).

b. Informatique embarquée et objets connectés

L'intérêt de cette thématique est de montrer aux élèves le fonctionnement des objets connectés, et **l'intérêt** de l'informatique embarquée. En effet, ici le but n'est

pas de créer un programme, mais bien de « intégrer de nouvelles possibilités [...] par simple modification ».

La grille des contenus et des capacités attendues (Annexe C) nous informe plus sur ce que les élèves devront savoir. On parle d'interface homme-machine, d'actionneurs et de capteurs. Nous avons donc choisi cette intégration avec des micros contrôleurs Arduino.

La dernière étape restera de créer un cours, ainsi que des activités sur cette thématique.

c. Internet

Pour la thématique d'internet, les élèves doivent apprendre le fonctionnement d'internet, car c'est « le moyen de communication principal » qui demeure aujourd'hui. En effet, que ce soit pour les cours (via l'ENT), pour discuter entre amis, tout passe par internet.

On retrouve au programme (Annexe D) les réseaux, les protocoles et les machines. Cela ressemble à ce que l'on a pu faire à l'université au semestre 3, mais toujours dans un intérêt de « comprendre les enjeux scientifiques et sociétaux de la science informatique ».

Le choix de logiciels pour cette thématique sera les logiciels Filius et Marionnet.

2. Numérique et sciences informatiques (NSI)

Comme la partie SNT, le programme complet est disponible dans la bibliographie (Ministère de l'Éducation nationale, 2019) et les notions importantes seront placées en annexe.

Cet enseignement est un enseignement au choix en première (4 h) et en terminale (6 h). Nous nous intéresserons qu'à la classe de première, car il n'y a eu aucune classe de terminale cette année.

L'intérêt de cette option est de prévoir une « poursuite d'études dans l'enseignement supérieur » en informatique. En effet, cet enseignement va développer la « démarche scientifique » des élèves grâce à des « activités de recherche ».

C'est un enseignement qui s'appuie sur les mathématiques, et notamment l'algorithmique vue dans cette matière en seconde.

Dans le préambule est également inscrite une liste de compétences qui devront être acquises à la fin de l'année. On y trouve notamment la validation de programmes, qui est similaire à la matière « Algorithmique avancée et complexité » enseignée à la fac.

Je vais maintenant analyser les différents éléments du programme que j'ai pu voir en classe.

a. Démarche de projet

Les élèves, par groupe de 2 à 4, auront pour but de mettre en œuvre les éléments acquis lors du cours via un projet. Ce projet, d'une durée totale d'un quart de l'année, sera défini sur un thème choisi par l'élève. Ce thème peut être en rapport avec une autre matière que la NSI.

Le professeur a pour rôle de vérifier le bon fonctionnement des groupes de travail, ainsi que le potentiel de réussite du projet.

b. Représentation des données

En informatique, les données sont représentées avec des 0 et 1, correspondant à un signal électrique ou non. On appelle cela un bit. Le programme est disponible en annexe (Annexe E).

Le programme, assez simpliste, correspond parfaitement à la matière « Numération et codage » dispensée à l'institut. On y retrouve les conversions, les tables booléennes, le code ASCII, et les flottants.

c. Langages et programmation

Il existe de nombreux langages de programmation, de différents types (impératif, objet, etc.). (Annexe F)

Dans cette partie, le langage choisi sera le Python. Il est notamment mentionné la « Spécification » de programme, c'est-à-dire indiquer les préconditions d'entrée et de sortie de programme, avec notamment l'ajout d'assertions.

d. Algorithmique

Dans cette partie, nous devons voir le fonctionnement des principaux algorithmes, tels que les algorithmes gloutons, ou de recherche dichotomique. De plus, nous voyons le « parcours séquentiel d'une liste » qui n'est pas si simple que cela à expliquer aux élèves.

Il faut également montrer les bonnes terminaisons des programmes, notamment avec l'utilisation de boucles While.

III. Les cours et exercices

Chaque semaine, l'enseignant m'envoyait le thème du cours, ainsi que les points à aborder. Dans certains cas, c'était à moi de réaliser des cours ou des activités pour les élèves.

1. Sciences numériques et technologie (SNT)

Lors de ma première arrivée en classe, les élèves étaient en train de travailler sur la représentation des données. J'ai donc pu les aider et leur expliquer les démarches pour convertir d'une base vers une autre. Je leur ai également expliqué comment fonctionnait le complément à deux⁴.

J'ai également mis en place plusieurs activités avec le professeur.

a. Arduino

Le but d'Arduino est de montrer l'intérêt et la présence de l'informatique de nos jours. En effet, les cartes Arduino sont des microcontrôleurs, et tous les appareils avec informatique embarquée en possèdent. C'est seul et avec un demi-groupe que j'ai pu réaliser les cours qui vont suivre ; le professeur étant dans la salle d'à côté.

Le premier cours réalisé s'intitule « Cours Arduino : Les bases » (Annexe I) et a été réalisé au mois de janvier pour introduire la notion « Informatique embarquée » du programme avec l'utilisation d'Arduino. Afin de le réaliser, je me suis inspiré du

⁴ https://fr.wikipedia.org/wiki/Compl%C3%A9ment_%C3%A0_deux

cours que nous disposions en Architecture des Ordinateurs (Garric, 2019). Ce cours commence par une brève histoire d'Arduino, ainsi que de la définition d'un microcontrôleur ; suivi par les caractéristiques de la carte Arduino UNO ; puis par la nomenclature du code.

Premier exercice du cours, faire clignoter une LED. Mais pour cela, il faut d'abord brancher une LED. Le cours comporte des notions vues par les élèves en physique au collège, telles que la loi d'Ohm, et le fonctionnement des LED (sens, résistance). Après avoir calculé la résistance nécessaire, et effectué les branchements, les élèves vont ensuite modifier le code d'exemple « Blink » afin de rediriger le clignotement vers la LED.

Cet exercice a plutôt bien été réalisé par les élèves, qui étaient en binômes. J'ai pu passer derrière chaque groupe pour répondre à leurs questions et leur apporter des précisions.

Le deuxième exercice n'a pas pu être réalisé avec tous les groupes, en raison d'un manque de temps et d'une mauvaise estimation du temps du cours. Cet exercice avait pour but de contrôler les clignotements avec un bouton.

À la fin de ce premier cours, j'ai pu discuter avec le professeur de mon ressenti seul devant une demie-classe. Nous avons également pu trouver des solutions afin d'améliorer le cours et de le rendre plus rapide, par exemple un branchement des plaquettes de travail déjà effectué avant l'arrivée des élèves. J'ai pu mettre en place ces modifications avec le groupe suivant, ce qui fut un succès.

Le deuxième cours s'intitule « Cours Arduino : Le LCD » (Annexe J) et a été réalisé deux semaines après le premier, dans les mêmes conditions de travail. Ce cours a été plus dans la pratique que le premier, car c'est ce qui motive le plus les élèves. Dans ce cours, j'ai pu apprendre aux élèves comment écrire des messages sur un écran LCD (Arduino, 2015) en utilisant des bibliothèques et des fonctions déjà écrites.

Ce deuxième et dernier cours ne conclut pas le travail sur Arduino, car un contrôle réalisé par le professeur a été effectué. Ce concours a été plutôt bien réussi par les élèves.

b. Réseau

Pour cette activité, j'ai utilisé les documents du livre (Katia Odiot, 2019) afin de créer une activité interactive. Il a fallu respecter le programme, c'est-à-dire montrer le fonctionnement du protocole IP, des réseaux, et du routage. Nous avons donc décidé de réaliser une activité avec des messages écrits sur un papier, représentant un message, et un numéro représentant une adresse IP.

Chaque élève dispose d'un numéro secret le correspondant, ainsi qu'un message et un destinataire. Les élèves ont dû trouver des méthodes pour échanger les messages entre eux en respectant les règles du réseau, notamment le fait de ne pouvoir parler qu'à une personne à la fois (voir un exemple en Annexe G).

Pendant l'activité, j'ai pu observer que les élèves étaient moins dispersés qu'en classe normale. L'activité manuelle leur a permis de quitter l'univers habituel de cours au tableau. Les élèves ont d'ailleurs particulièrement bien réussi cette activité.

La semaine suivante, les éléments vus pendant l'activité ont été mis en application dans un TP sur le logiciel Filius (Annexe H). Les élèves ont dû paramétrer les options réseau des machines virtuelles afin de leur permettre de communiquer entre elles. Ils ont aussi paramétré un serveur DNS et un serveur Web.

2. Numérique et sciences informatiques (NSI)

Cette matière a été la plus avancée, due au fait que les élèves sont en première, et comme vu dans l'analyse, le programme est de haut niveau. De plus, 3 élèves de la licence sont venus.

a. Arduino

Un projet global était de mettre en place un réseau de capteurs. Pour cela, la première étape a été de récupérer les données des capteurs. Les élèves ont donc codé cette carte Arduino afin de recevoir les informations des capteurs, et les transmettre sur un certain port.

Comme pour les SNT, j'ai présenté mon cours, mais avec une approche différente. En effet, les élèves sont plus avancés et les cours sont plus longs. J'ai donc

utilisé les plaquettes déjà faites du prof, contenant une carte Arduino, un écran LCD, un capteur de distance, et un cercle de LEDs NeoMatrix.

Afin de pouvoir utiliser ces modules ensemble, j'ai décidé de les séparer en trois petites activités réparties dans un PowerPoint (Annexe K). De plus, la présentation était accompagnée de simulations via Tinkercad⁵ (Annexe L).

À la fin de l'activité, cela a débouché sur le projet que devaient faire les élèves. Je me suis occupé des élèves ayant choisi de faire leur projet autour d'Arduino.

L'un des groupes avait comme projet de mesurer le pH avec Arduino, et d'envoyer mes données sur un serveur. De même pour un autre groupe, mais avec un capteur de pression et de température.

b. Réseau

Avec les élèves de première, le réseau a été traité de manière différente. Ils ont appris comment créer un serveur, avec les cours de l'enseignant. Ce serveur a donc permis de faire la liaison avec l'activité précédente sur Arduino.

De plus, ils ont travaillé avec le logiciel Marionnet afin d'apprendre le fonctionnement des réseaux, et notamment apprendre comment fonctionnent le protocole IP, les masques de sous-réseaux, le protocole DNS, et les serveurs Web.

Afin d'évaluer leurs compétences, nous avons réalisé une évaluation que j'ai pu corriger (Annexe M). Cette évaluation (1 h) n'a pas été bien réalisée, notamment à cause d'un oubli d'envoi de fichier. En effet, certains élèves n'ont tout simplement pas enregistré leur travail, sachant que les travaux enregistrés sont automatiquement stockés sur le PC de l'enseignant.

À part cela, le cours a été compris pour la grande majorité des élèves.

c. Python

Le langage de développement est le Python. C'est pourquoi les élèves ont dû réaliser une activité pour revoir le langage. Cette activité (Annexe N) a été créée par

⁵ Tinkercad est un programme de simulation gratuit qui s'exécute dans un navigateur Web, connu pour sa simplicité et sa facilité d'utilisation. Le programme est détenu par Autodesk.

trois élèves en licence informatique qui ont choisi de venir faire une activité au lycée de XXXXXXXX.

Cette activité est très globale et regroupe tous les thèmes importants en programmation informatique. En effet, on y retrouve la représentation des données, le traitement de données en tables, l'interaction homme-machine, ainsi que de la programmation. Ce sont des thèmes que l'on retrouve dans le programme (Ministère de l'Éducation nationale, 2019).

Afin de compléter l'activité, j'ai réalisé un PowerPoint sur les listes en python, afin de mettre en image ce type de donnée qui paraît complexe auprès des élèves. La présentation était composée de comparaisons avec ce que connaissent les élèves, afin de leur faire retenir plus aisément le cours. Extraits en Annexe O.

Finalement, l'activité Bases de programmation a été évaluée, et j'ai pu la corriger (). Les exercices ont été plutôt bien réussis, malgré quelques élèves qui n'ont rien écrit. Pourtant nous étions 4 à tourner autour des élèves pour les aider.

IV. Conclusion du stage

Nous avons pu voir dans ce rapport les différentes techniques utilisées afin d'enseigner l'informatique à des élèves de seconde et de première. Les différentes notions du programme font réfléchir à des cours et exercices de plus en plus intéressants et éducatifs.

De plus, ce stage m'a permis dans un premier temps de comprendre le métier d'enseignant. J'ai pu apprendre comment se prépare un cours, quels sont les documents dont dispose l'enseignant, et comment captiver l'attention des élèves.

J'ai également pu découvrir les nouvelles matières d'informatique au lycée. Cela m'a fait rendre compte que certains enseignants peuvent être perdus, car ils n'ont pas été formés à cela.

Le stage m'a également montré qu'il est possible d'enseigner dans le respect, et que les classes non assidues sont peu, voire inexistantes. Dans le cas de XXXXXXX, je n'ai aucun problème avec les élèves, et j'ai même été heureux de voir que si l'enseignement était bien réalisé, alors les élèves étaient heureux du résultat.

Bibliographie

Architecture des ordinateurs : Arduino [En ligne] = Cours Arduino 1 / aut. Garric Nicolas // Moodle. - Octobre 2019. - 10 Avril 2020. - https://moodle.univ-jfc.fr/pluginfile.php/77378/mod_resource/content/0/Cours-Arduino1.pdf.

LiquidCrystalDisplay [En ligne] = LCD / aut. Arduino // Arduino. - 17 Août 2015. - 10 Avril 2020. - <https://www.arduino.cc/en/Tutorial/LiquidCrystalDisplay>.

Programmes et ressources en numérique et sciences informatiques [En ligne] / aut. Ministère de l'Éducation nationale // éducol. - 22 Janvier 2019. - 28 Mars 2020. - <https://eduscol.education.fr/cid144156/nsi-bac-2021.html>.

Programmes et ressources en sciences numériques et technologie [En ligne] / aut. Ministère de l'Éducation nationale // éducol. - 22 Janvier 2019. - 28 Mars 2020. - <https://eduscol.education.fr/cid143713/snt-bac-2021.html>.

Sciences numériques et technologie 2nde [Livre] = SNT 2nde / aut. Katia Odiot Dominique Lagrula. - [s.l.] : Hachette Éducation, 2019. - 2019 : pp. 12-15. - 9782017866053.

Annexe A : Organigramme

XXXXX

Annexe B : Tableau des notions de programmation

Contenus	Capacités attendues
Affectations, variables Séquences Instructions conditionnelles Boucles bornées et non bornées Définitions et appels de fonctions	Écrire et développer des programmes pour répondre à des problèmes et modéliser des phénomènes physiques, économiques et sociaux.
Exemples d'activités	
- Illustrer ces notions par des activités liées aux différents thèmes du programme.	

(Ministère de l'Éducation nationale, 2019)

Annexe C : Tableau des attentes sur l'informatique embarquée

Contenus	Capacités attendues
Systèmes informatiques embarqués	Identifier des algorithmes de contrôle des comportements physiques à travers les données des capteurs, l'IHM et les actions des actionneurs dans des systèmes courants.
Interface homme-machine (IHM)	Réaliser une IHM simple d'un objet connecté.
Commande d'un actionneur, acquisition des données d'un capteur	Écrire des programmes simples d'acquisition de données ou de commande d'un actionneur.
Exemples d'activités	
<ul style="list-style-type: none">- Identifier les évolutions apportées par les algorithmes au contrôle des freins et du moteur d'une automobile, ou à l'assistance au pédalage d'un vélo électrique.- Réaliser une IHM pouvant piloter deux ou trois actionneurs et acquérir les données d'un ou deux capteurs.- Gérer des entrées/sorties à travers les ports utilisés par le système.- Utiliser un tableau de correspondance entre caractères envoyés ou reçus et commandes physiques (exemple : le moteur A est piloté à 50 % de sa vitesse maximale lorsque le robot reçoit la chaîne de caractères « A50 »).	

(Ministère de l'Éducation nationale, 2019)

Annexe D : Tableau des attentes sur internet

Contenus	Capacités attendues
Protocole TCP/IP : paquets, routage des paquets	Distinguer le rôle des protocoles IP et TCP. Caractériser les principes du routage et ses limites. Distinguer la fiabilité de transmission et l'absence de garantie temporelle.
Adresses symboliques et serveurs DNS	Sur des exemples réels, retrouver une adresse IP à partir d'une adresse symbolique et inversement.
Réseaux pair-à-pair	Décrire l'intérêt des réseaux pair-à-pair ainsi que les usages illicites qu'on peut en faire.
Indépendance d'internet par rapport au réseau physique	Caractériser quelques types de réseaux physiques : obsolètes ou actuels, rapides ou lents, filaires ou non. Caractériser l'ordre de grandeur du trafic de données sur internet et son évolution.

Exemples d'activités

- Illustrer le fonctionnement du routage et de TCP par des activités débranchées ou à l'aide de logiciels dédiés, en tenant compte de la destruction de paquets.
- Déterminer l'adresse IP d'un équipement et l'adresse du DNS sur un réseau.
- Analyser son réseau local pour observer ce qui y est connecté.
- Suivre le chemin d'un courriel en utilisant une commande du protocole IP.

(Ministère de l'Éducation nationale, 2019)

Annexe E : Représentation des données

Contenus	Capacités attendues	Commentaires
Écriture d'un entier positif dans une base $b \geq 2$	Passer de la représentation d'une base dans une autre.	Les bases 2, 10 et 16 sont privilégiées.
Représentation binaire d'un entier relatif	Évaluer le nombre de bits nécessaires à l'écriture en base 2 d'un entier, de la somme ou du produit de deux nombres entiers. Utiliser le complément à 2.	Il s'agit de décrire les tailles courantes des entiers (8, 16, 32 ou 64 bits). Il est possible d'évoquer la représentation des entiers de taille arbitraire de Python.
Représentation approximative des nombres réels : notion de nombre flottant	Calculer sur quelques exemples la représentation de nombres réels : 0.1, 0.25 ou 1/3.	0.2 + 0.1 n'est pas égal à 0.3. Il faut éviter de tester l'égalité de deux flottants. Aucune connaissance précise de la norme IEEE-754 n'est exigible.
Valeurs booléennes : 0, 1. Opérateurs booléens : and, or, not. Expressions booléennes	Dresser la table d'une expression booléenne.	Le ou exclusif (xor) est évoqué. Quelques applications directes comme l'addition binaire sont présentées. L'attention des élèves est attirée sur le caractère séquentiel de certains opérateurs booléens.
Représentation d'un texte en machine. Exemples des encodages ASCII, ISO-8859-1, Unicode	Identifier l'intérêt des différents systèmes d'encodage. Convertir un fichier texte dans différents formats d'encodage.	Aucune connaissance précise des normes d'encodage n'est exigible.

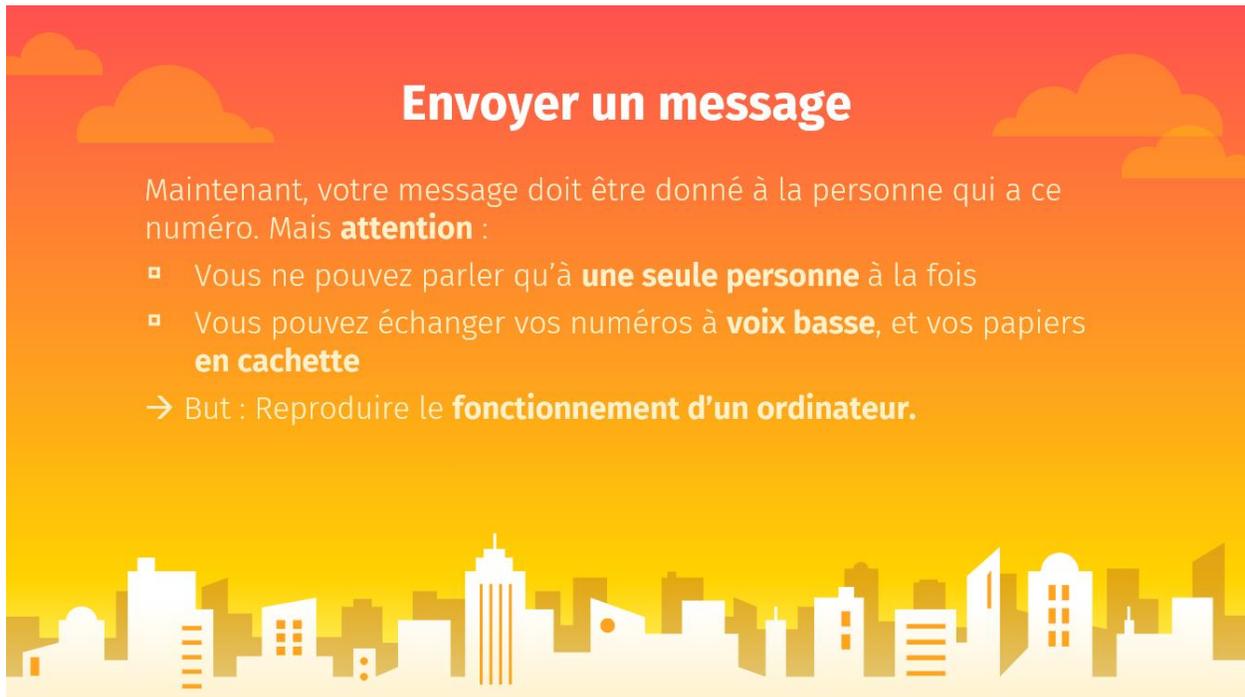
(Ministère de l'Éducation nationale, 2019)

Annexe F : Langages et programmation

Contenus	Capacités attendues	Commentaires
Constructions élémentaires	Mettre en évidence un corpus de constructions élémentaires.	Séquences, affectation, conditionnelles, boucles bornées, boucles non bornées, appels de fonction.
Diversité et unité des langages de programmation	Repérer, dans un nouveau langage de programmation, les traits communs et les traits particuliers à ce langage.	Les manières dont un même programme simple s'écrit dans différents langages sont comparées.
Spécification	Prototyper une fonction. Décrire les préconditions sur les arguments. Décrire des postconditions sur les résultats.	Des assertions peuvent être utilisées pour garantir des préconditions ou des postconditions.
Mise au point de programmes	Utiliser des jeux de tests.	L'importance de la qualité et du nombre des tests est mise en évidence. Le succès d'un jeu de tests ne garantit pas la correction d'un programme.
Utilisation de bibliothèques	Utiliser la documentation d'une bibliothèque.	Aucune connaissance exhaustive d'une bibliothèque particulière n'est exigible.

(Ministère de l'Éducation nationale, 2019)

Annexe G : Extrait activité réseaux



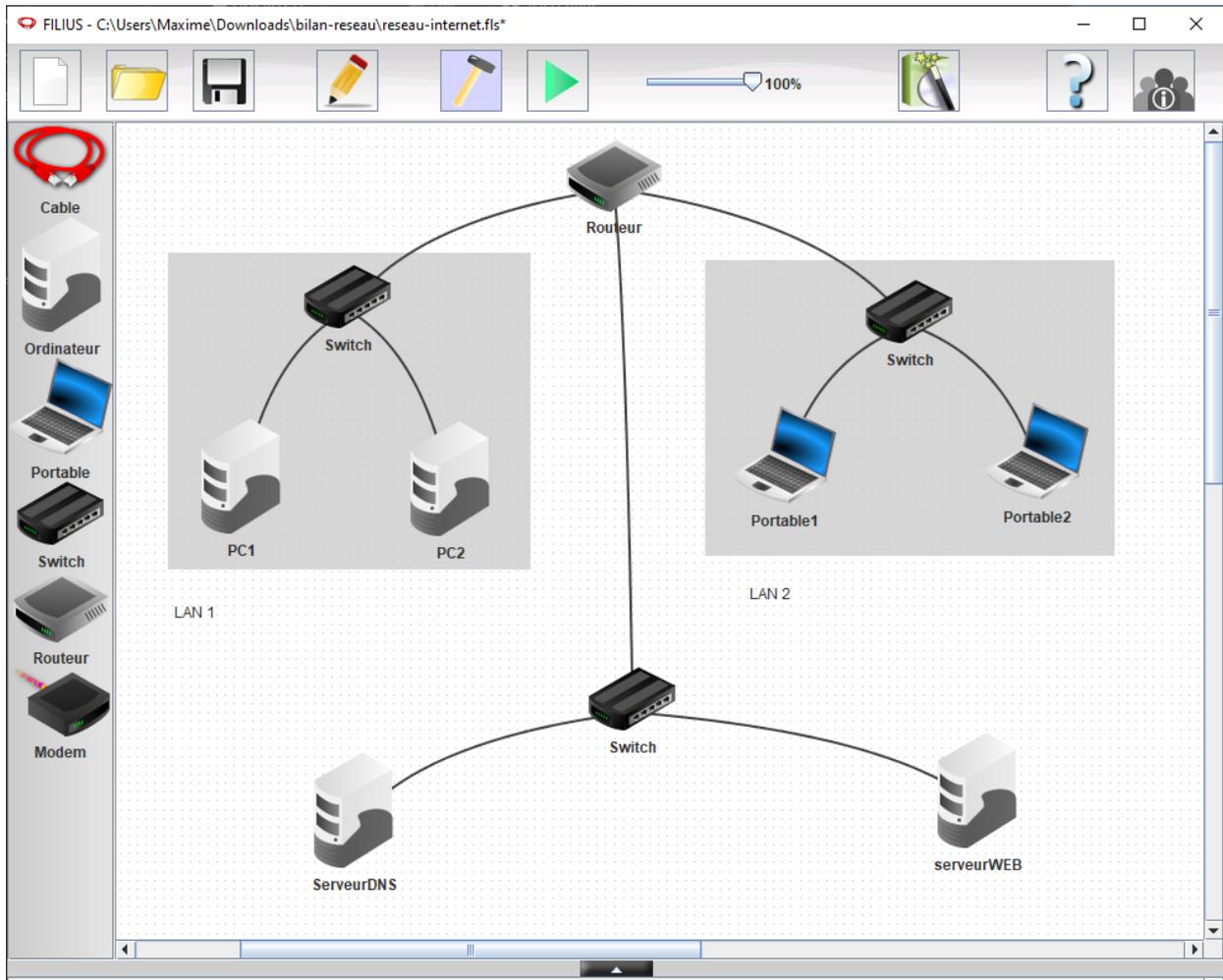
Envoyer un message

Maintenant, votre message doit être donné à la personne qui a ce numéro. Mais **attention** :

- Vous ne pouvez parler qu'à **une seule personne** à la fois
- Vous pouvez échanger vos numéros à **voix basse**, et vos papiers **en cachette**

→ But : Reproduire le **fonctionnement d'un ordinateur**.

Annexe H : Le logiciel Filius



Annexe I : Cours Arduino - Les bases

Solutions RH SUD Cours Arduino Les bases *hello*

Arduino est né en 2005 par 5 amis, en Italie. C'est un micro-contrôleur, c'est-à-dire un petit ordinateur mini d'un écran, d'un clavier, d'un souris... Le but est de faire de l'informatique embarquée. Les vidéos, écrans dans le bus ou objets connectés

Fonctionnement:

Capteurs → Programme → Actionneur

- Arduino UNO: Mémoire: 32Ko RAM: 2Ko 14 E/S numériques, 6 E analogiques

MES NOTES

Solutions Rh Sud
Abit - Tél : 05 63 43 25 47
Toulouse - Tél : 05 34 29 19 01
Organisme de formation enregistré sous le numéro 13 03541 83
Cet enseignement est validé par l'Agence de l'éducation

Datadeck
Opérateur de formation
OPCF

Les broches (pin en anglais) servent à brancher les composants

Le code

- définition des variables
- initialisation
- boucle

On doit indiquer à quoi servira l'E/S: est-ce une entrée ou une sortie?
pinMode(numéro, INPUT/OUTPUT);

Pour lire une broche, on utilise digitalRead(numéro);
Pour écrire sur une broche, on utilise digitalWrite(numéro, état);

Exemple: Blink

- Ouvrir fichier exemple
- Choisir le port USB
- Téléverser

Branchement des LEDs

La tension de la LED est de $U_L = 2V$ donc la tension de la résistance doit être de $U_R = 3V$

Le courant délivré est de $I = 10mA$

$R = \frac{U}{I} = 300 \Omega$

MES NOTES

Solutions Rh Sud
Abit - Tél : 05 63 43 25 47
Toulouse - Tél : 05 34 29 19 01
Organisme de formation enregistré sous le numéro 13 03541 83
Cet enseignement est validé par l'Agence de l'éducation

Datadeck
Opérateur de formation
OPCF

Annexe J : Cours Arduino - Le LCD

Solutions RH SUD Cours Arduino: Le LCD

- Afficheur alphanumérique (16x2)
- Possible de contrôler chaque pixel de chaque rectangle

- Câblage déjà fait sur la plaquette

Code:

Inclure la bibliothèque: `#include <LiquidCrystal.h>`
 Créer l'initialiser l'écran: `LiquidCrystal lcd(7, 6, 5, 4, 3, 2);`
 Setup: `RS E D4 D5 D6 D7`
 → indiquer la taille de l'écran `lcd.begin(16, 2);`

Fonctions:

- `lcd.print(str);`
- `lcd.setCursor(col, line);`
- `lcd.clear();` → efface et règle la position en 0,0

Voir plus: arduino.cc/en/Reference/LiquidCrystal

MES NOTES

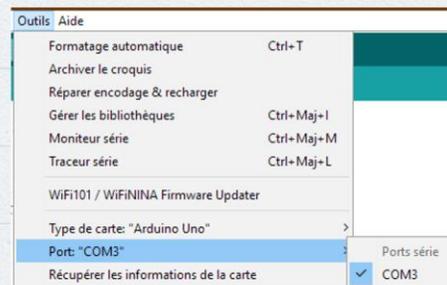
Solutions Rh Sud
Abit - Tél : 05 63 43 25 47
Toulouse - Tél : 05 34 29 19 01
Organisme de formation enregistré sous le numéro 13 03541 83
Cet enseignement est validé par l'Agence de l'éducation

Datadeck
Opérateur de formation
OPCF

Annexe K : Détecteur de distance

COMMENT UTILISER L'IDE ARDUINO ?

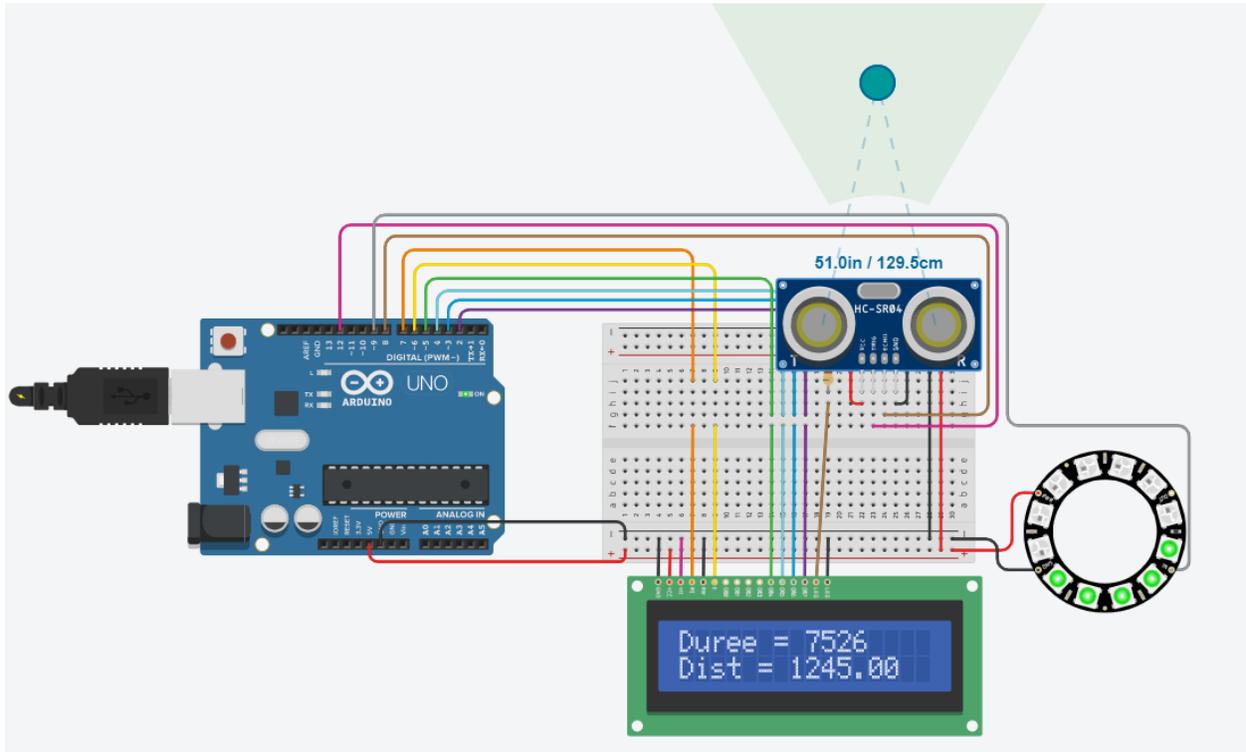
- x Le code se vérifie avec le bouton 
- x Il faut vérifier que le bon port est sélectionné
- x Le code est envoyé avec bouton 



```
5 // Inclure la librairie LiquidCrystal
6 #include <LiquidCrystal.h>
7
8 //Initialisation du lcd
9 LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
10
11 //---le setup executer une fois au démarrage
12 void setup() {
13   //initialisation serie
14   Serial.begin(9600);
15   Serial.print("Initialisation lcd");
16   // Configuration de l'afficheur 16 caractères en 2 lignes
17   lcd.begin(16, 2);
18   // Ecriture de la ligne 1
19   lcd.print("initialisation:");
20   delay(2000);
21   //remise a zero de l'ecran
22   lcd.clear();
23 }
```

```
25 //La boucle principale
26 void loop() {
27
28   Serial.print("Bonjour");
29   Serial.println();
30   lcd.setCursor(0, 0); //Positionnement du curseur
31   lcd.print("Test ligne 1 ");
32   lcd.setCursor(0, 1);
33   lcd.print("Test ligne 2 ");
34   delay(20000);
35   lcd.clear();
36 }
```

Annexe L : Tinkercad



Annexe M : Evaluation Marionnet

Fiche d'intervention

Pour connaître la configuration réseau d'un machine, on peut utiliser ifconfig.

Pour faire des modifications qui tiennent , il faut

1. ouvrir le fichier réseau avec la commande nano /etc/network/interfaces
2. faire les modifications
3. enregistrer (CTRL + X) , Y, puis entrée

Les machines m2, m5 et m6 sont correctement configurée, il n'y a pas de modifications à faire .

Avec marionnet :(ALT + F2) marionnet

Ouvrir le fichier entrainement1.mar

Tout démarrer

Répondre sur ce fichier . enregistrer régulièrement

1pt

I) Combien de réseaux compte ce 'domaine' ?

2

7pb

II) Compléter le tableau ci-dessous - 0,5

machine	m1	m2	m3	m4	m6
ethX	eth0	eth0	eth0	eth0	eth0
address	192.168.2.10	192.168.2.2	1.1	1.20	1.1
netmask	24	24	24	24	24
gateway	2.154	2.254	2.254	1.254	1.234
dns-nameservers	2.2.8.8	2.2.8.8	—	—	—

1pt 1pt 1pt 1pt 1pt

réseau - 0,5
cible + réseau + gw - 2,5

machine	m5	m5
ethX		1
address	2.254	1.254
netmask	24	24
gateway	2.154	1.154
dns-nameservers	—	—

1pt

1pt

3
2,5pb

III) Lister les problèmes rencontrés à votre avis

3pb

IV) Configuration

Faire les modifications / vérifier les machines m1,m3 et m4 pour qu'elle puissent communiquer entre elles , ainsi qu'avec la machine m6.

Quand l'épreuve sera terminée, arrêter toutes les machines et enregistrer le projet marionnet !

NOM Prénom	I		II			III	IV			Total	Appréciation
	Q 1	DNS 0,5	Mask 0,5	@ 2	GW 1	Q 1	m1 1	m3 1	m4 2		
	0	0,5	0,5	2	1	0	0	0	0,5	4,5	2 réseaux
	0	0	0	0	0	0	1	0	0	1	fichiers vide
	1	0,5	0,5	1,5	1	0	0	0	1,5	6	@ m4 avant problème III?
	1	0,5	0,5	2	1	0	1	0	2	8	Le gw sert à communiquer entre les réseaux. III pour fichier vide
	0	0	0	0	0	0	0	0	0	0	
	0	0,5	0,5	2	1	1	0	0	0	5	et le I? manonnet? III
	0	0,5	0,5	2	1	0	0	1	1,5	6,5	2 réseaux
	0	0	0	0	0	0	0	0	0	0	fichier vide
	1+	0,5	0,5	1,5	1	0	0	0	0	4,5	@ m3 eth1 incorrect III esp vrai mais où pb?
	1	0,5	0,5	1,5	1	0	0	0	0	5,5	△ Erreurs de frappe mask @ m2 incorrect
	1	0,5	0,5	1,5	0,5	0	1	0	2	7	@ m3 eth1 incorrect que m3 → 154 256
	1	0,5	0,5	2	1	0,5	0	0	0	5,5	manonnet? III pourquoi?
	1	0,5	0,5	2	0,5	0	0	0	1	5,5	pourquoi les deux switches sont reliés? gw
	1	0,5	0,5	2	1	0	0	0	0	5	III? manonnet!
	1	0,5	0,5	2	1	1	1	0	0,5	7,5	cable m4? m2? gw à modifier aussi m4
	1	0,5	0,5	2	1	1	0	0	1	7	TB, manque config ip
	0	0	0	0	0	0	0	0	1	1	fichier vide
	1	0,5	0,5	1,5	1	1	0	0	1,5	7	@ m3 eth1 incorrect pb gw + réseau
	0	0	0	0	0	0	0	0	0	0	Fichier vide + manonnet non réalisé
	1	0,5	0,5	2	0,5	1	0	0	0	5,5	fichier manonnet vide
	1+	0,5	0,5	2	0,5	0	0	0	0	4,5	gw différents pour éditer fichier, déplacé avec souris
	1	0,5	0,5	2	0,5	0	0	0	0	4,5	gw m1 faux
	1	0,5	0,5	2	1	1	0	0	0	6	TB, manque manonnet III
	0	0	0	0	0	0	0	0	0	0	fichiers vide

moy = 4,4 (5,3)

Annexe N : Extraits de Bases de programmation

1) Interaction avec la console: print et input

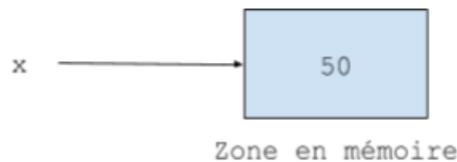
print est un élément très important en programmation: il permet d'afficher un ou plusieurs éléments dans la console.

Exemples d'utilisation:

```
print("Hello World!")
print(50)
print(9.80665)
print("Plusieurs", "éléments")
```

```
Hello World!
50
9.80665
Plusieurs éléments
```

print est l'outil idéal quand il s'agit de débogger un programme.



On dit que x est une **référence** ou que x **pointe** vers la zone en mémoire.

Chaque variable a un type. Le type d'une variable est une indication sur la nature des données stockées dans la zone en mémoire: il permet à python de correctement interpréter ce qui s'y trouve.

Maintenant petite explication sur ce qu'il se passe:

```
i = 0
while i < 100:
    print("Bonjour les gens!")
    i = i + 1
```

Mot clé **while**: **Déclaration** de la boucle

Condition $i < 100$: **Condition de continuation** de la boucle

Instructions en **bleu**: **Corps** de la boucle.

Annexe O : Extrait du cours sur les Listes

Structure de données

- Tableau d'éléments
- Numérotation de 0 à n-1

0	Anaïs
1	Sophie
2	Léo
3	Jade
...	...
n-1	Enzo

Cellule n°3

- Comparaison avec le tableur Excel

	A	B	C
1			
2			
3			
4			
5			
6			
7			

Cellule n°B2

2

En Python

Soit L une liste :

- `L = []` → Création d'une liste vide
- `L.insert(i, x)` → Ajoute l'élément x à l'indice i
- `L.append(x)` → Ajoute l'élément x à la fin de la liste
- `L.pop(i)` → Supprime l'élément à l'indice i
- `len(L)` → Renvoie la taille de la liste
- `L[i]` → Renvoie l'élément à l'indice i
- `L[i] = x` → Remplace l'élément à l'indice i par x

5

Annexe P : Evaluation Python

NOM	Bases de programmation					Données		TOTAL	Appréciation
	Ex 4	Ex 5	Ex 6	Ex 7	Ex 8	Ex 1	Ex 2		
	1	3	2	2	3	2	2	15	
	0,5	1,5	0,5					2,5	18 ans mineur!
⚠	0,5	3	1					4,5	18 ans mineur? Ex 6? print(somme)! + input
	1	2,5	1,5	2	1			8	erreur texte input ex6 TB sinon
	1	3	2	2	2			10	penser à nommer les variables les fonctions ont des paramètres et return
	1	3						1	
⚠	0,5	1						1,5	18 ans mineur?
	1	2,5	2	2	0,5			8	0,1mm = 0,0001m
	1	1,5						2,5	
	1	3 ⁺	2	1				7 ⁺	ex 7: manque affectation pour ok
	1	2						3	
	0,5	3	1,5	2	1			7,5	18 paramètres? ex 6: remplir input
	1	0,5	2	2	0,5			6	Boucle sans fin 1mm ≠ 1m
	1	3	2	2				8	TB
	1	1,5	2	2				6,5	for feuille = feuille * 2 do while ???
	1	3	2	1				7	tableau: 1 dimension
	1	2	2	2	0,5			7,5	nom des variables incohérent
	1	3	1					5	ex 4: a > 18 et a == 18 idem a >= 18 ok
	1	2,5	1,5	2	0,5			7,5	nb pairs en 5 for: mauvais retour + incrémentation
	1	3	2	1				7	liste: aléatoire ok mais pair & modulo!!
⚠	1	1	1					3	ex 6: copie colle
	1	2	0,5	2				5,5	nb impairs: j = 0? Non début à 1 ex 6: affiche str de pays liste
	1	1	0,5					2,5	() print + 1mm = 0,001m ex: for!!
	0,5	1	0,5	0				2	18 mineur?
⚠	1	1	1					3	ex 6: COPIE COLLE
	1	2	1	2				6	ex 5: feuille = feuille * 2