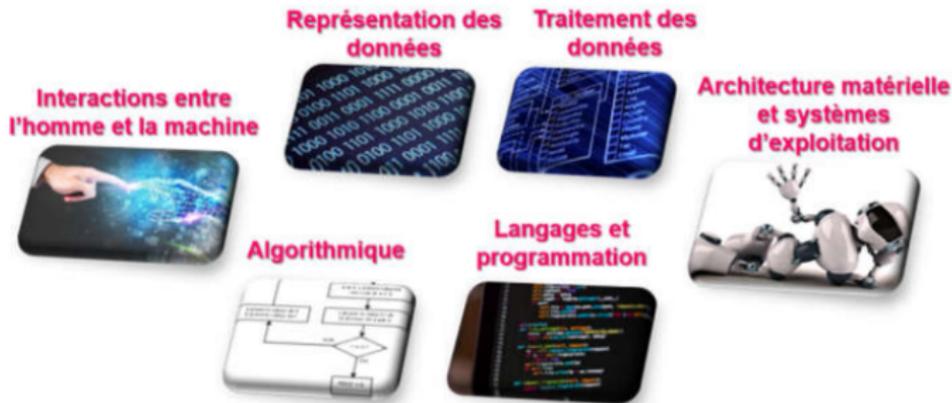
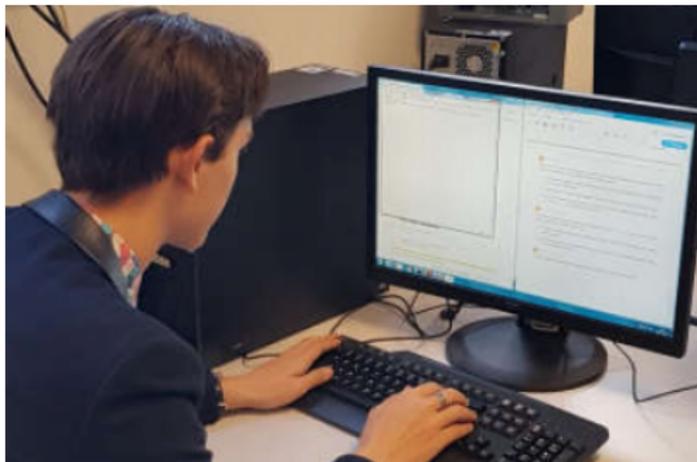


A.B. & D.T.



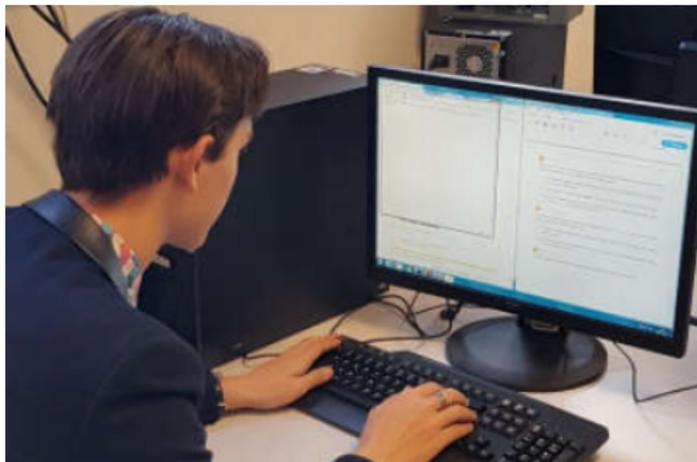
Les aptitudes requises

- Appétence pour les sciences, l'informatique et le numérique,



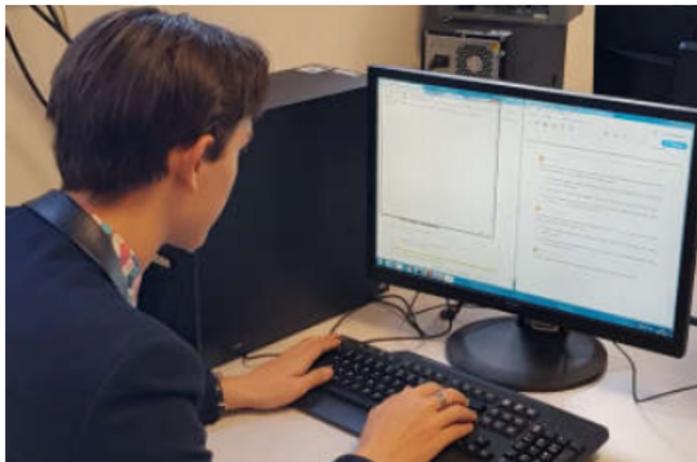
Les aptitudes requises

- Appétence pour les sciences, l'informatique et le numérique,
- autonomie, capacité de rédaction, esprit de synthèse, goût du travail collaboratif,



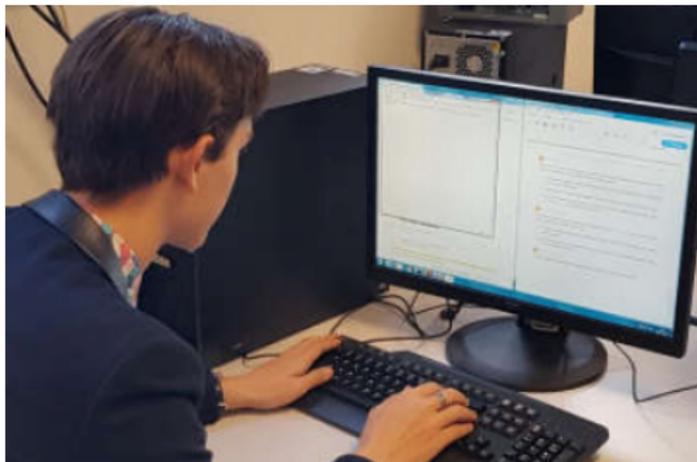
Les aptitudes requises

- Appétence pour les sciences, l'informatique et le numérique,
- autonomie, capacité de rédaction, esprit de synthèse, goût du travail collaboratif,
- aimer résoudre des problèmes,



Les aptitudes requises

- Appétence pour les sciences, l'informatique et le numérique,
- autonomie, capacité de rédaction, esprit de synthèse, goût du travail collaboratif,
- aimer résoudre des problèmes,
- être prêt à étudier des notions assez abstraites.



Objectifs de l'enseignement

- Préparer les élèves à la poursuite d'étude en enseignement supérieur.
- S'appropriier les concepts et les méthodes qui fondent l'informatique :
 - Les données
 - Les algorithmes
 - Les langages
 - Les machines



- L'histoire de l'informatique.



- L'histoire de l'informatique.



- La représentation et le traitement de données.



- L'histoire de l'informatique.



- La représentation et le traitement de données.



- Les algorithmes.

Algorithme de recherche de l'indice du minimum d'une liste (minimum)

Entrée : Une liste non triée de n éléments, l'indice j de début de recherche

Sortie : L'indice du minimum de cette liste dans l'intervalle $[j; n-1]$.

```
1 liste ← n éléments
2 j ← indice de début de recherche
3 indice ← indice du min
4 Min ← liste[j]
5 pour i allant de j+1 à n-1 faire
6   si liste[i] < Min alors
7     Min ← liste[i]
8     indiceMin ← i
9 Renvoyer indiceMin
```

- La programmation dans le langage évolué :  python™

```
def tri_insertion(liste):
    for i in range(1,len(liste)): #pour parcourir les éléments
        tour à tour
        print(i)
        while (liste[i]<liste[i-1] and i>0): #TQ l'élément est <
            à la valeur précédente
            liste[i-1], liste[i] = liste[i], liste[i-1] #on
            permute
            i = i-1 #en remontant les éléments i à 1 jusqu'à ce
            qu'il soit à la bonne place
        print(liste)
L=[3,2,18,9,6]
tri_insertion(L)
```

- La programmation dans le langage évolué :  python™

```
def tri_insertion(liste):  
    for i in range(1,len(liste)): #pour parcourir les éléments  
        tour à tour  
        print(i)  
        while (liste[i]<liste[i-1] and i>0): #TQ l'élément est <  
            à la valeur précédente  
            liste[i-1], liste[i] = liste[i], liste[i-1] #on  
            permute  
            i = i-1 #en remontant les éléments i à 1 jusqu'a ce  
            qu'il soit à la bonne place  
        print(liste)  
L=[3,2,18,9,6]  
tri_insertion(L)
```

- Recherche et tri de données,



- La programmation dans le langage évolué :  python™

```
def tri_insertion(liste):
    for i in range(1,len(liste)): #pour parcourir les éléments
        tour à tour
        print(i)
        while (liste[i]<liste[i-1] and i>0): #TQ l'élément est <
            à la valeur précédente
            liste[i-1], liste[i] = liste[i], liste[i-1] #on
            permute
            i = i-1 #en remontant les éléments i à 1 jusqu'a ce
            qu'il soit à la bonne place
        print(liste)
L=[3,2,18,9,6]
tri_insertion(L)
```

- Recherche et tri de données,



- Interactions entre l'homme et la machine sur le Web



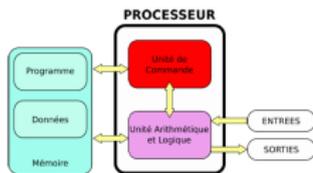
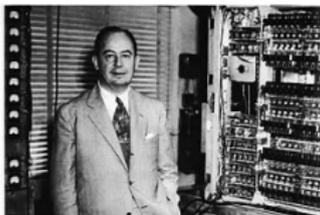
- Réseaux informatiques



- Réseaux informatiques

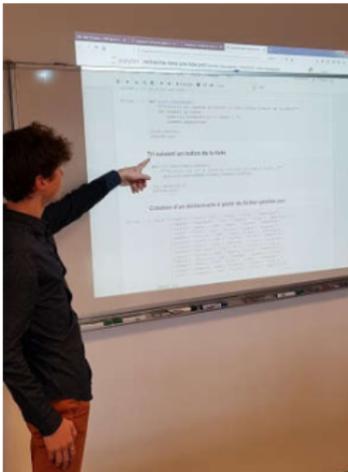


- Architectures matérielles et systèmes d'exploitation



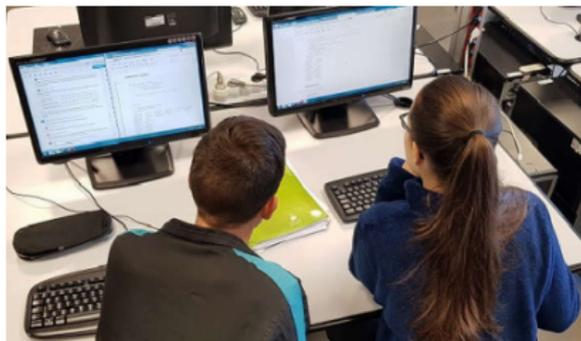
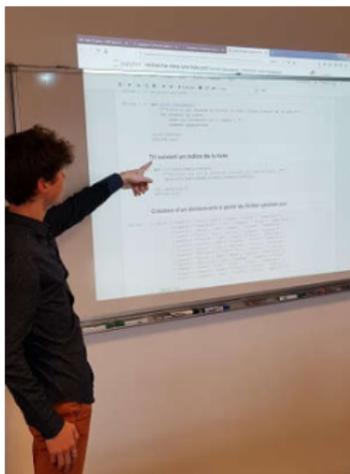
L'organisation pédagogique

- Cours et TD



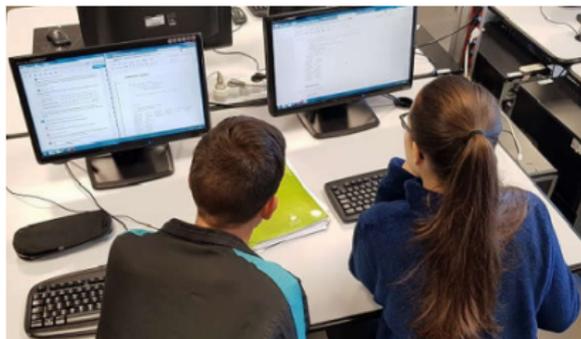
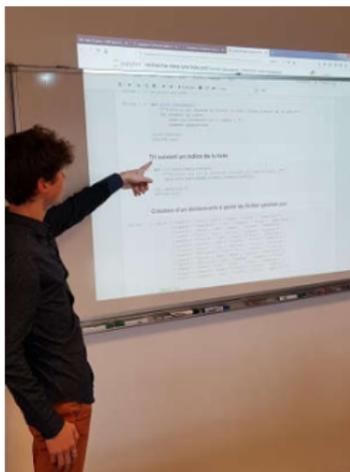
L'organisation pédagogique

- Cours et TD
- Activités pratiques individuelles ou en binômes



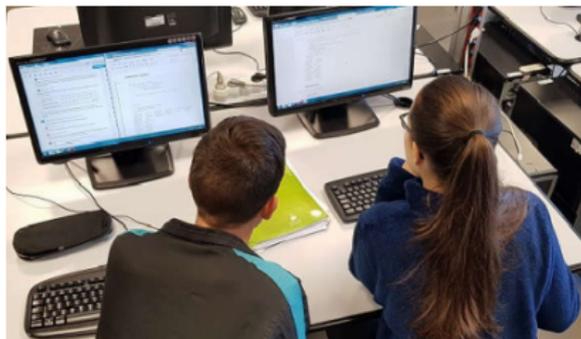
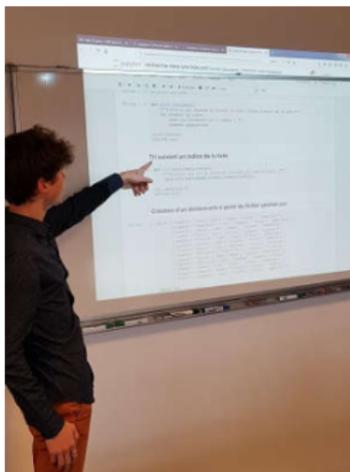
L'organisation pédagogique

- Cours et TD
- Activités pratiques individuelles ou en binômes
- Mini projets ou projets réalisés en groupe de 2 à 4 élèves



L'organisation pédagogique

- Cours et TD
- Activités pratiques individuelles ou en binômes
- Mini projets ou projets réalisés en groupe de 2 à 4 élèves
- La démarche de projet est privilégiée.



- Réalisation d'une page WEB pour présenter la section NSI
- Réaliser un programme de changement de base.

```
def bin_dec(codeBinaire:str):  
    L = []  
    for element in codeBinaire: #convertit la chaine de caractères  
        #binaires en une liste de 8 bits  
        L.append(element)      #L[0] est le poids fort  
  
    dec = 0  
    for i in range(8):  
        dec = dec+int(L[i])*2**(7-i) #permet de calculer la  
            #valeur décimale de chaque bit en fonction de sa  
            #position  
    return str(dec) #retourne une chaine de caractères  
        #correspondant à la somme décimale des 8 bits
```

- Traiter des données pour la gestion d'un Zoo.
- Faire un programme permettant de quantifier l'énergie et le coût lié à l'utilisation d'un appareil électrique.



- Gérer les statistiques d'un joueur de NBA.

2019-2020 SAISON STATS																		
CLASSEMENT	JOUEUR	EQUIPE	M	MI	PPM	RPM	PPRM	MPM	EFF	FG%	3PL%	%F	DÉSACTIVE	DÉF	IPM	CPM	BP	FP
1	 James Harden	HOU	56	56	34.9	6.4	7.3	36.8	32.3	44.0	36.0	86.4	1.1	5.3	1.7	0.9	4.4	3.4
2	 Bradley Beal	WAS	51	51	30.3	4.4	6	35.9	25.3	45.8	33.5	84.4	1	3.4	1.1	0.4	3.4	2.3
3	 Trae Young	ATL	56	56	29.8	4.3	9.4	35.4	27.1	44.1	36.3	86.2	0.5	3.8	1.3	0.1	4.7	1.7
4	 Glenn Robinson III	MIL	53	53	29.7	13.7	5.8	38.8	34.9	55.3	31.7	62.2	2.3	11.3	1	1.1	3.7	3.1
5	 Damian Lillard	POR	54	54	29.5	4.4	7.9	37	28.3	46.0	39.3	88.9	0.5	3.8	1	0.3	2.9	1.7
6	 Luka Dončić	DAL	48	48	28.6	9.4	8.8	32.9	30.7	46.3	31.9	76.7	1.4	8.1	1	0.2	4.3	2.5
7	 Russell Westbrook	HOU	49	49	27.6	7.9	7.2	36.1	26.9	47.3	25.1	78.2	1.7	6.2	1.7	0.3	4.4	2.6



- Décryptage d'un texte

Le texte encodé avec un décalage $d = 2$ (pour vérification) :

gp 1967, n'cefgokg htpeckug fghkpkv n'kphqtocvkswg eqoog « nc uekgpeg fw vtckvgogpv tvckvppgn, pqvcoogpv c n'ckfg fg ocejkpgu cwwqocvkswu, fg n'kphqtocvkswg, eqpukftgg eqoog ng uwrtrqv fg eqppckuucpegu fcpu ngu fqockpgu uekgpvkhkswg, geppqokswg gv uqekcn. »



Les poursuites d'études

La spécialité NSI permet d'évaluer son goût pour l'informatique avant de s'engager dans des études supérieures dans l'informatique ou dans le domaine scientifique :

La nouveauté de 2021 : création des classes préparatoires MP21 (Mathématiques, Physique, Ingénierie et Informatique)

