Distribution d'un jeu de cartes, division euclidienne et preuve d'un algorithme.

Atelier L1-09

I - Présentation du groupe EIEM

- Mathématiques et Informatique
- Réflexion sur l'utilisation de l'informatique en maths puis en SNT et NSI
- Publication en cours sur cette activité (revue Radix)
- Stage LaTeX Pays de la Loire

II - Contexte

Extrait du programme de 1ère NSI :

Contenus	Capacités attendues	Commentaires
Tris par insertion, par sélection	Décrire un invariant de boucle qui prouve la correction [...]	La terminaison de ces algorithmes est à justifier.
Recherche dichotomique dans un tableau trié	Montrer la terminaison [...] à l'aide d'un variant de boucle.	La preuve de la correction peut être présentée par le professeur.

Contenus	Capacités attendues	Commentaires
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.

II - Contexte

- Notions difficiles et fortement liées aux mathématiques
- Diversité des profils d'élèves en 1NSI
- Objectif de départ :

  Trouver une situation simple pour introduire la notion de preuve d'un algorithme.

III - Précisions sur les notions

La **correction** d'un algorithme est la preuve qu'il remplit son rôle : qu'à partir d'entrées vérifiant les pré-conditions il produit correctement des sorties vérifiant les post-conditions.

- La **correction partielle** vérifie que l'algorithme produit le bon résultat.
- La **terminaison** vérifie qu'il produit un résultat en un temps fini.

III - Précisions sur les notions

Si l'algorithme présente une boucle, on utilise un invariant pour montrer la correction partielle.

Un **invariant** de boucle est une expression logique, dépendante des variables de la boucle, qui est vraie au début et à la fin de chaque itération.

Cette dernière notion est proche du raisonnement par récurrence.

IV - L'activité

- Durée : 1h
- Niveau 1ère NSI
- Activité débranchée
- Manipulation de jeux de cartes
- Par groupes de 3 à 5 élèves

IV - L'activité

Première distribution

- Bonne / maldonne ?

  ...
- L'algorithme :

  ...

IV - L'activité

Première distribution

- Post-conditions :

  Chacun des joueurs a le même nombre de cartes, le nombre de cartes du talon est strictement inférieur au nombre de joueurs.

- L'algorithme :

  « Tant qu'il reste plus de cartes que de joueurs, distribuer une carte à chaque joueur. »

IV - L'activité

Seconde distribution interrompue : vers la notion d'invariant

- On interrompt la distribution :

  « Peut-on savoir si, pour l'instant, la distribution est bonne, ou s'il y a déjà maldonne ? »

Jusqu'ici tout va bien...

IV - L'activité

Invariant de boucle

- On retient la proposition :

  « Chaque joueur a le même nombre de cartes. »

IV - L'activité

Suite de la distribution

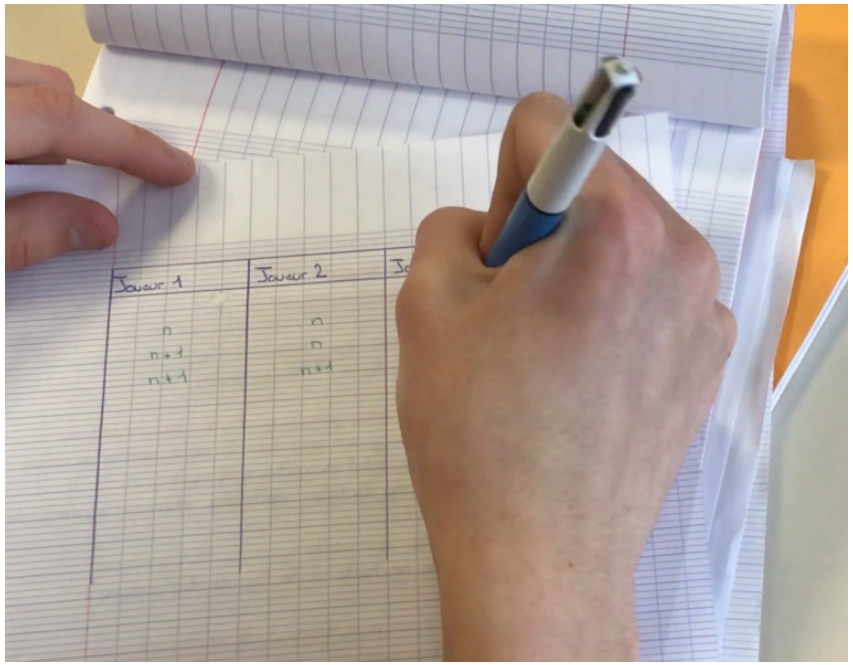
- Les groupes changent de place
  Subtiliser une carte (ou pas)
- Effectuer un tour de distribution en plus :

  « Peut-on savoir si la distribution est toujours bonne, sans toucher aux cartes ? »

IV - L'activité

Suite de la distribution

- Tracer l'exécution lors d'un tour de la boucle



IV - L'activité

Formalisation de la démonstration :

- On a montré que, si l'invariant est vrai avant un tour de boucle, alors il reste vrai après.
- On remonte le raisonnement : l'invariant est vérifié au départ quand aucun joueur n'a de carte.
- On peut donc conclure que l'invariant reste vrai après chaque itération de la boucle. Il est donc encore vrai à la fin de celle-ci.

- Reste à montrer qu'elle se termine...

IV - L'activité

Propriété :
P_n vraie pour tout n entier naturel.

Initialisation :
P_o est vraie.

Hérédité :
$\forall k \in \mathbb{N}, P_k vraie \Rightarrow P_{k+1} vraie.$

Conclusion
$\forall n \in \mathbb{N}, P_n vraie.$

Propriété : Chaque joueur a le même nombre de cartes.

Initialisation :
Avant la distribution aucun joueur n'a de carte.

Hérédité :
Après k tours, les joueurs ont le même nombre de cartes $\Rightarrow$ après $k+1$ tours les joueurs ont le même nombre de cartes.

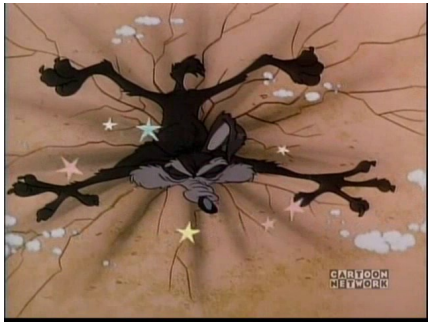
Conclusion
A chaque tour chacun des joueurs a le même nombre de cartes.

V - Analyse

- Plusieurs réalisations de l'activité par plusieurs enseignants en 1 NSI depuis 3 ans
- Séance dialoguée
- Poursuite, transposition à d'autre situations difficile

VI - Poursuites

Preuve de terminaison



- Montrer que la boucle s'arrête
- Utiliser un variant de boucle

Un **variant de boucle** est une fonction des variables de la boucle :

- à valeurs entières positives ;
- qui décroît strictement à chaque itération.

VI - Poursuites

D'autres exemples

- Multiplication par sommes successives
- Recherche d'un maximum dans un tableau
- Crible d'Ératosthène

Algorithmes repris en mathématiques expertes :

- Écriture d'un entier dans une base
- Algorithme d'Euclide

VI - Poursuites

Multiplication par sommes successives

Pré-conditions : a et b sont des entiers

- mettre résultat à 0 ;
- répéter b fois : ajouter a à résultat

Post-condition : résultat contient a $\times$ b

VI - Poursuites

Multiplication par sommes successives

Pré-conditions : a et b sont des entiers

- mettre résultat à 0 ;
- répéter b fois : ajouter a à résultat

Post-condition : résultat contient a $\times$ b

Invariant : résultat est égal au produit de a par le nombre de répétitions déjà effectuées.

VI - Poursuites

Recherche d'un maximum dans un tableau

Pré-condition : tab est un tableau de nombres

- maxi prend la valeur du premier élément de tab
- Parcourir les valeurs de tab (de la seconde à la dernière) :
  si la valeur courante de tab est supérieure à maxi,
  mettre maxi à cette valeur

Post-condition : maxi contient la valeur maximale de tab

VI - Poursuites

Recherche d'un maximum dans un tableau

Pré-condition : tab est un tableau de nombres

- maxi prend la valeur du premier élément de tab
- Parcourir les valeurs de tab (de la seconde à la dernière) :
  si la valeur courante de tab est supérieure à maxi,
  mettre maxi à cette valeur

Post-condition : maxi contient la valeur maximale de tab

Invariant : maxi est le maximum de la partie du tableau déjà parcourue.

VI - Poursuites

Crible d'Ératosthène

Pré-conditions : tab est un tableau contenant les entiers de 2 à n

- Pour chaque nombre du tableau :
- Parcourir tab du nombre suivant jusqu'au dernier
  si l'élément parcouru est un multiple du nombre : le supprimer

Post-condition : tab ne contient que des nombres premiers

VI - Poursuites

Crible d'Ératosthène

Pré-conditions : tab est un tableau contenant les entiers de 2 à n

- Pour chaque nombre du tab :
- Parcourir tab du nombre suivant jusqu'au dernier
  si l'élément parcouru est un multiple du nombre : le supprimer

Post-condition : tab ne contient que des nombres premiers

Invariant : La partie du tableau, du début jusqu'au dernier nombre traité ne contient que des nombres premiers.

VI - Poursuites

Algorithme d'Euclide

Pré-conditions : a et b deux nombres entiers naturels ($b \neq 0$)

- On effectue la division euclidienne de a par b, puis celle du diviseur par le reste obtenu et ainsi de suite jusqu'à obtenir un reste nul ;
- On note, pour $i \geqslant 1$, q_i et r_i les quotients et restes obtenus.

Post-condition : Le dernier reste non nul est égal à $PGCD(a\ ;\ b)$

VI - Poursuites

Algorithme d'Euclide

Pré-conditions : a et b deux nombres entiers naturels ($b \neq 0$)

- On effectue la division euclidienne de a par b, puis celle du diviseur par le reste obtenu et ainsi de suite jusqu'à obtenir un reste nul ;
- On note, pour $i \geqslant 1$, q_i et r_i les quotients et restes obtenus.

Post-condition : Le dernier reste non nul est égal à $PGCD(a; b)$

Invariant : $PGCD(a\ ;\ b) = PGCD(r_{i+1}\ ;\ r_i)$

VI - Poursuites

Décomposition

Pré-conditions : b un nombre entier supérieur ou égal à 2 et N un nombre entier naturel.

- On effectue les divisions euclidiennes successives de N, puis des quotients obtenus, par b jusqu'au rang k tel que $q_k = 0$;
- On note $q_0 = N$ puis, pour $i > 1$, q_i et r_i les quotients et restes obtenus.

Post-condition : $N = b^{k-1} \times r_k + b^{k-2} \times r_{k-1} + \cdots + b \times r_2 + r_1$

VI - Poursuites

Décomposition

Pré-conditions : b un nombre entier supérieur ou égal à 2 et N un nombre entier naturel.

- On effectue les divisions euclidiennes successives de N, puis des quotients obtenus, par b jusqu'au rang k tel que $q_k = 0$;
- On note $q_0 = N$ puis, pour $i > 1$, q_i et r_i les quotients et restes obtenus.

Post-condition : $N = b^{k-1} \times r_k + b^{k-2} \times r_{k-1} + \cdots + b \times r_2 + r_1$

Invariant : $N = b^i \times q_i + b^{i-1} \times r_i + \cdots + b \times r_2 + r_1$