

Activité : Jeu de rôle (Primaire et Collège)

Objectifs :

- comprendre la notion de composant d'un ordinateur
- réaliser qu'il se passe beaucoup de choses à l'intérieur d'un ordinateur quand un programme, même très simple, est exécuté
- découvrir le bus de communication
- appréhender la notion de variable informatique (espace de la mémoire possédant un nom et un contenu qui peut évoluer pendant l'exécution)

Durée : 1h15 à 1h30

Matériel :

- Étiquettes : processeur, mémoire, clavier, écran, bus
- Clavier : fiche instruction correspondant au programme
- Mémoire : une feuille plastifiée pour chaque variable (préciser leur nom)
- Écran : un tableau (ou une feuille affichée)
- Bus : une ardoise
- Processeur : selon les cas une petite fiche pour effectuer les calculs
- 1 fiche instructions par élève
- 1 programme/tableau par élève (il y a 4 programmes)
- Craies, feutres, chiffons, brosses, selon les rôles

Niveaux : CM1, CM2, 6ème, 5ème

Prérequis : notions d'instruction et de séquence d'instructions.

Phase : Introduction

Durée : 5'

Activité et consignes : voir "Activité préliminaire"

Phase : Présentation du schéma de l'architecture

Durée : 5'

Activité et consignes :

Dégager les cinq éléments que l'on va étudier pour la séance : le clavier, l'écran, le processeur, la mémoire, le bus.

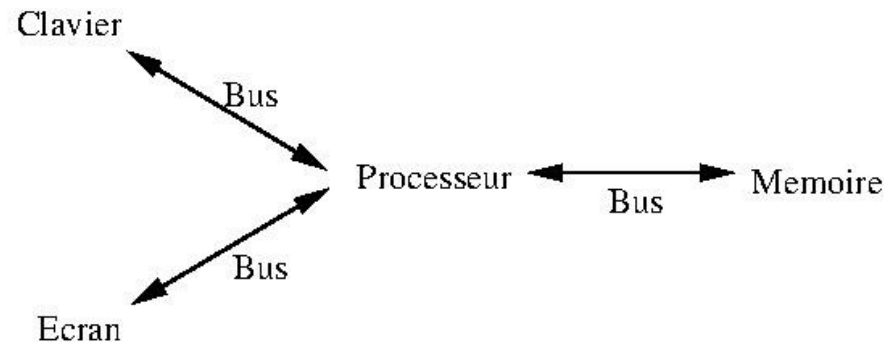
Définir chaque élément :

- le processeur est le chef d'orchestre (c'est lui qui commande les autres éléments et leur dit ce qu'ils doivent faire),
- la mémoire est le lieu où sont stockées les valeurs des variables,
- le clavier est composé de touches permettant de saisir des caractères,
- l'écran permet d'afficher des informations
- et le bus (les fils) permet à tous ces éléments de communiquer entre eux.

Définir également les mots suivants :

- variable : espace de la mémoire qui contient une valeur et qui a un nom. On peut imaginer le tiroir d'une commode avec un nom sur le tiroir, ou un casier d'étagère avec une étiquette. La valeur contenue dans une variable peut évoluer.
- caractère : touche du clavier (une lettre ou un chiffre, ou un autre signe, comme par exemple , = (etc.

Représenter au tableau par un schéma comme ci-dessous les interactions entre ces différents éléments.



Le processeur peut communiquer avec le clavier, l'écran et la mémoire, mais ceux-ci ne peuvent pas communiquer entre eux. Les communications ne se font jamais directement, c'est obligatoirement le bus qui sert d'intermédiaire. Comme c'est le processeur qui est le chef d'orchestre, c'est toujours lui qui demande au bus de porter un message à un autre composant, et éventuellement de revenir avec une réponse. Le bus ne transmet qu'un seul message à la fois, et il revient tout de suite vers le processeur pour attendre l'ordre suivant.

Phase : Description des rôles

Durée : 10'

Activité et consignes :

Professeur : Nous allons faire un jeu qui va nous permettre de comprendre comment ces cinq éléments communiquent entre eux. Le processeur est responsable de l'exécution des programmes qu'on lui fournit. Les programmes sont des séries d'instructions. Le processeur ne comprend que des instructions pour lesquelles il a été préprogrammé. Le processeur organise l'exécution d'une instruction à la fois, puis il oublie ce qui vient de se passer et passe à l'instruction suivante. Souvent, l'exécution d'une seule instruction comprend plusieurs étapes.

Lire la liste des instructions (au moins pour les deux premiers programmes) et les distribuer aux élèves.

Bien définir le rôle de chacun :

- **le processeur (P)** donne des ordres au bus et, à travers lui, aux autres composants. C'est un rôle qui nécessite beaucoup d'autonomie.

Pour les programmes 3 et 4, il peut lire, écrire et effacer l'ardoise et il dispose de petits papiers type post-it et d'un crayon (car il possède une toute petite mémoire qui lui permet de faire des calculs).

$$\boxed{} \times 10 + \boxed{} = \boxed{}$$

$$\boxed{} \cdot \boxed{} = \boxed{}$$

Entre les programme 2 et 3, expliquer le rôle et l'utilisation de cet outil à partir d'exemples, expliquer quelles sont les opérations que sait faire le processeur.

- **le bus (B)** transporte une ardoise entre les autres protagonistes, sur laquelle il ne peut ni lire, ni écrire, ni effacer. Le bus montre ce qui est écrit sur l'ardoise au public.
- **le clavier (C)** suit les instructions de la petite fiche secrète pour chaque programme. Il peut effacer puis écrire sur l'ardoise. Dans le jeu de rôle, le clavier et l'utilisateur sont confondus pour simplifier.
- **la mémoire (M)** stocke la valeur de différentes variables dans ses « cases », qui sont représentées par des feuilles plastifiées (une feuille pour chaque variable), sur lesquelles elle peut effacer puis écrire. On a 2 possibilités : en entrée → lire puis effacer, en sortie → effacer puis écrire. C'est un rôle qui nécessite beaucoup d'autonomie, lorsqu'il y a plusieurs variables dont les valeurs changent au cours de l'exécution (programmes 3 et 4).
- **l'écran (E)** utilise un tableau pour l'affichage sur lequel il peut écrire (mais pas effacer dans les exemples proposés). Il peut lire sur l'ardoise, puis effacer.

Phase : exécution du premier programme (saisir et afficher un caractère).

Cinq élèves sont désignés pour réaliser le premier programme.

Le clavier reçoit une feuille secrète décrivant ce qu'il doit faire :

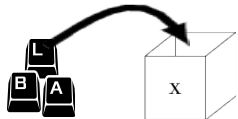

Le clavier tape un caractère.

Placer les différents personnages (sauf le bus) dans un espace de la classe. On utilise des étiquettes avec leur rôle pour les identifier. Pour le bus, on peut épingler l'étiquette sur les vêtements de l'élève.

Professeur : "Nous allons montrer ce qui se passe à l'intérieur de l'ordinateur quand un utilisateur tape (par exemple) le caractère "A" sur le clavier et qu'il s'affiche à l'écran."

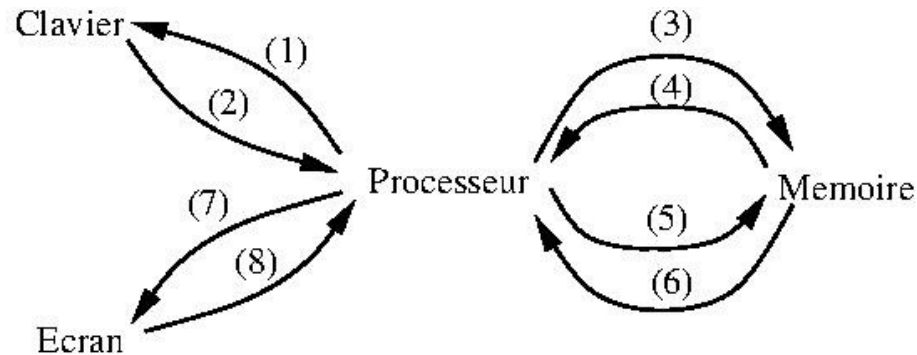
Distribuer le programme/tableau d'exécution aux élèves (sauf le bus, qui est trop occupé), qui le complètent. Ce tableau d'exécution (ci-dessous) permet de noter tout ce qui se passe au fur et à mesure de l'exécution du programme. Chaque ligne numérotée (éventuellement divisée en sous-lignes) correspond à l'exécution d'une instruction. Une seule instruction peut correspondre à plusieurs actions des composants, chaque action étant représentée par une sous-ligne. Il est préférable de remplir le tableau au crayon à papier. Pour permettre au public de remplir le tableau, les élèves qui jouent les composants disent soigneusement à voix haute tout ce qu'ils font, et le bus montre bien son ardoise au public. Pour une variante plus simple, il est possible de projeter et remplir ce tableau d'exécution collectivement au tableau.

Saisir et afficher un caractère

	Programme	Actions	Mémoire X
1			
2			

La machine est allumée, ce qui signifie que tous les composants P, B, C, E et M sont prêts à travailler. De plus, B est à côté de P.

Le schéma ci-dessous peut-être dessiné au tableau au fur et à mesure de l'exécution pour matérialiser les déplacements du bus, c'est-à-dire les interactions des composants.



P commence à lire le programme.

P prend connaissance de l'instruction 1 :



P dit à B : va demander à C d'écrire sur l'ardoise le caractère saisi

B y va (1)

C efface l'ardoise et écrit sur l'ardoise le caractère saisi

B rapporte l'ardoise à P (2)

L'enseignant fait verbaliser comment remplir le tableau : dans la colonne "Actions" on écrit "Demander clavier : A" (par exemple), et dans la colonne "Mémoire x", on n'écrit rien, puisque pour le moment on n'y a rien mis.

P dit à B : va montrer l'ardoise à M pour qu'il stocke dans la variable "x" (autrement dit qu'il recopie sur la feuille portant ce nom) ce qui est écrit sur l'ardoise

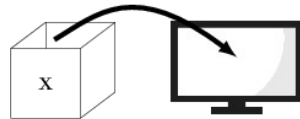
B y va et lui dit stocke dans la variable "x" ce qui est écrit sur l'ardoise (3)

M stocke dans la variable "x" ce qui est écrit sur l'ardoise (lit/efface)

B revient vers P (4)

Dans le tableau, cela se traduit par : dans la colonne "Actions", on écrit "Stocker X : A" et dans la colonne "Mémoire X", on écrit (par exemple) "A"
Fin de l'exécution de l'instruction 1

P prend connaissance de l'instruction 2 :



P dit à B : va demander à M d'envoyer la valeur de la variable "x" (autrement dit d'effacer l'ardoise et de recopier dessus ce qui est écrit sur la feuille portant ce nom)

B y va et lui demande d'envoyer la valeur de la variable "x" (5)

M communique la valeur de la variable "x"

B revient vers P (6)

Dans le tableau, cela se traduit par : on écrit "Demander X : A" dans la colonne "Actions", et dans la colonne "Mémoire x", il y a toujours écrit "A", puisque cette valeur n'a pas changé

P dit à B : va montrer l'ardoise à E pour qu'il recopie sur le tableau ce qui est écrit sur l'ardoise

B y va (7)

E recopie sur le tableau ce qui est écrit sur l'ardoise

B revient vers P (8)

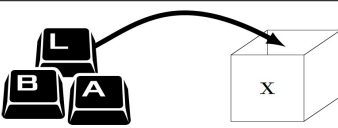
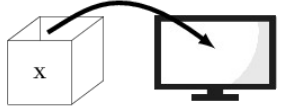
Dans le tableau, cela se traduit par : on écrit "Afficher : A" dans la colonne "Actions", et dans la colonne "Mémoire x", il y a toujours écrit "A", puisque cette valeur n'a pas changé

Fin de l'exécution de l'instruction 2

C'est terminé.

Pour validation, on demande à C si la lettre écrite au tableau est bien celle qu'il avait tapée. Le tableau rempli devrait ressembler à ceci :

	Programme	Actions	Mémoire X
--	-----------	---------	-----------

1		Demander clavier : A	
		Stocker X : A	A
2		Demander X : A	A
		Afficher : A	A

Commentaires :

- Dans cette phase, il est important de bien souligner que le processeur décompose l'exécution de chaque instruction en plusieurs étapes.
- Pour éviter les confusions, nous recommandons de remplir la colonne "Actions" en se plaçant du point de vue du processeur. Par exemple, pour la première case, on dira "demander au clavier quel caractère a été tapé; la réponse du clavier est : A", plutôt que "taper A".

Phase : Exécution du deuxième programme (saisir et afficher des initiales)

Durée : 10'

Activité et consignes :

Remarques concernant les programmes 2 à 4 :

- Le schéma d'exécution devient trop compliqué pour être dessiné au tableau.
- Distribuer les tableaux d'exécution correspondant aux programmes à tous les élèves
- Les tableaux d'exécution à remplir contiennent les instructions, et les sous-lignes sont déjà préparées.
- Les tableaux d'exécution déjà remplis sont fournis pour l'enseignant.
- D'autres élèves jouent le rôle des composants. Les élèves qui jouent le rôle du processeur de la mémoire et du bus sont très occupés, ils n'ont probablement pas le temps de remplir le tableau, c'est pourquoi il est important que les rôles tournent.

Remarques concernant le programme 2 :

- Le programme précédent est répété deux fois, il s'agit d'une situation de réinvestissement.
- La notion de mise à jour d'une variable qui contenait déjà une valeur est rencontrée pour la première fois. Dans l'activité, lorsque la mémoire doit modifier le contenu d'une variable, elle efface ce qui est sur la feuille correspondante et écrit à la place la nouvelle valeur.

Phase : Exécution du troisième programme (saisir stocker et afficher une année)

Durée : 10'

Activité et consignes :

Nouveautés :

- Il y a deux variables (le rôle de la mémoire devient plus compliqué)
- mémoire temporaire du processeur (le rôle du processeur devient plus compliqué)
- les nombres 0 et 10 sont utilisés sans être tapés au clavier : ils sont connus du processeur. Ce sont des valeurs constantes qui ne dépendent pas de l'utilisateur
- calculs (et détailler l'ordre des calculs $3 \times 10 + 1 = (3 \times 10) + 1$ si les priorités opératoires ne sont pas acquises)

Phase : Exécution du quatrième programme (saisir stocker et afficher secrètement un code de 4 caractères)

Durée : 10'

Activité et consignes :

Nouveautés :

- Le caractère * est affiché sans avoir été tapé au clavier : il est connu du processeur (c'est une constante, qui ne dépend pas de l'utilisateur).

- on rencontre une opération appelée la concaténation qui consiste à mettre bout à bout deux mots ("chaînes de caractères"). Par exemple "Bon"."jour" donne "Bonjour".
- Cette opération est très importante en informatique. Il est possible d'insister sur la différence chiffre-calcul et lettre-concaténation.

Phase : Bilan

Durée : 5'

Activité et consignes :

- Nombre d'instructions par seconde dans un vrai ordinateur. Présence d'un ventilateur pour rafraîchir les composants qui chauffent
- Le bus est indispensable : presque tout passe par lui.
- notion de variable informatique
- le processeur a une petite mémoire qui fait gagner beaucoup de temps car elle est tout près de lui (imaginer que la mémoire est dans la classe à côté)
- horloge/cadencement : tout le monde va au rythme de processeur, comme le chef d'orchestre donne la cadence aux musiciens. Il y a des processeurs plus rapides que d'autres.

Organisation et matériel : Oral collectif

Phase : Évaluation

Demander aux élèves par groupe de 5 d'écrire leur propre programme et le jouer. Par exemple:

- un programme qui effectue l'addition de deux chiffres (nombres entre 0 et 9) entrés successivement au clavier et affiche le résultat : on peut doter le processeur d'un nouveau post-it "... + ... = ..." et le langage d'une nouvelle instruction $X+Y \rightarrow Y$. Si on demande en plus d'afficher chacun des deux chiffres, il faudra aussi prendre en compte une contrainte importante : si une valeur entrée au clavier est directement affichée à l'écran sans être d'abord stockée dans une variable, elle est perdue pour la suite du programme et ne peut plus être utilisée dans un calcul.
- un programme qui calcule le double d'un chiffre entré au clavier et affiche le résultat.