



Projet Jeu du pendu

Objectifs

À l'issue de ce module, vous serez capable, à partir d'un descriptif textuel, de comprendre, analyser, et implémenter les concepts algorithmiques fondamentaux :

- Variables (type et portée)
- Conditions
- Boucles
- Fonctions

Modalités

- Durée du projet : 3 jours.
- Ce projet sera réalisé en îlots

Description du projet

- Le but de ce projet est de coder le jeu du pendu.
- Afin de vous faciliter la tâche, le projet a été découpé en itérations successives.

Informations et consignes générales

Les différentes itérations vont suivre un schéma répétitif.

Modalités

- Travail en autonomie puis en groupe

- Production individuelle

Consignes

Pour chaque itération,

1. Exercice ordinateur éteint:
 - a. Créer sur papier l'organigramme individuellement.
 - b. Chaque îlot confronte leur idée pour en proposer une solution.
 - c. Chaque îlot présente leur idée à tout le monde.

À la fin de la présentation, il sera possible de faire des remarques.

2. Exercice sur ordinateur

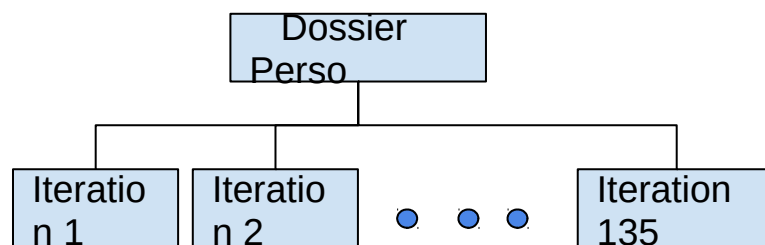
Chaque personne réalise le code en fonction de son organigramme sur Processing.

Livrables

- ☐ Organigrammes
- ☐ Code Processing (.pde) sur votre dossier personnel.

Il n'y a pas d'attente particulière au niveau graphique, tout peut se faire dans la console.

Ces livrables seront stockés dans vos dossiers personnels itération par itération.



Ressources

- <https://processing.org/reference/>
- <https://openclassrooms.com/fr/courses/1397856-processing>
- https://www.cs.bham.ac.uk/~cxp291/ri/processing_cheat_sheet_english.pdf
- [https://fr.wikipedia.org/wiki/Organigramme_\(logique_et_informatique\)](https://fr.wikipedia.org/wiki/Organigramme_(logique_et_informatique))
- <https://cacao.com/>

Etape 1 : Une lettre à deviner

Objectifs de l'activité

- Se familiariser avec les types "char" en Java
- Produire un logiciel avec saisie utilisateur
- Formaliser un logigramme simple

Cahier des charges

Une lettre inconnue du joueur doit être devinée. Cette lettre sera “codée en dur” (manuellement) par le programmeur. Le joueur possède un nombre illimité de tentatives pour deviner la lettre.

À chaque tentative, le joueur saisit une lettre, le programme affiche la lettre proposée par l'utilisateur et signale si la lettre est celle à deviner ou non. Le jeu s'arrête lorsque la lettre est devinée.

Etape 2 : Un mot à trouver

Objectifs de l'activité

- Se familiariser avec les types “String” en Java
- Produire un logiciel avec saisie utilisateur
- Formaliser un logigramme

Cahier des charges

Cette fois le but est de retrouver un mot en entier. Ce mot sera choisi par le programmeur. Le mot sera caché et chaque lettre remplacée par un “_”.

Chaque fois que l'utilisateur devine correctement une (ou plusieurs) lettre(s), le programme remplace le “_” correspondant par la (ou les) bonne(s) lettre(s). Une fois que le mot a été deviné entièrement, le jeu est terminé.

Etape 3 : Le pendu final

Objectifs de l'activité

- Se familiariser avec les types “String” en Java
- Produire un logiciel avec saisie utilisateur
- Formaliser un logigramme

Prérequis

- Les logigrammes de l'étape précédente
- Les codes Processing de l'étape précédente

Cahier des charges

Cette fois-ci nous incluons également la gestion du nombre de tentatives. L'utilisateur aura un nombre de tentative prédéfini pour trouver le mot.

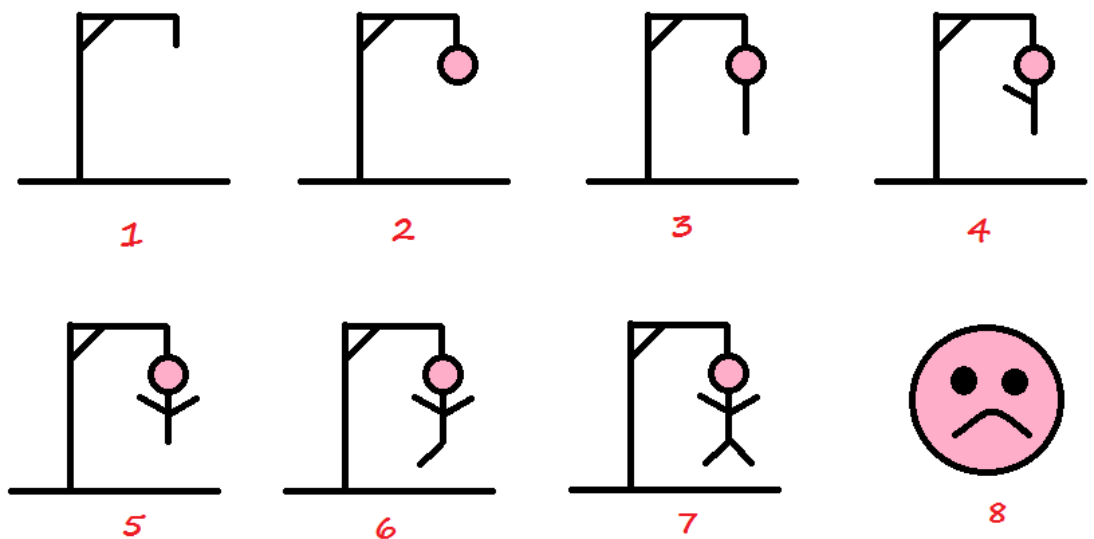
Il faudra donc afficher le mot à deviner sous forme de “_”, mais aussi le nombre de tentatives restantes, la lettre proposée par l'utilisateur et la réponse de la machine (“Cette

lettre n'est pas présente dans le mot" / "Cette lettre est présente 3 fois dans le mot".

Le jeu se termine une fois le mot deviné ou la limite de proposition atteinte.

Pour aller plus loin

- Le mot est maintenant inconnu du programmeur. Le programme sélectionne un mot au hasard dans un fichier texte contenant une liste de mot défini.
- Le programme affichera en plus toutes les lettres de l'alphabet, une fois qu'une lettre est utilisée, celle-ci n'est plus affichée au tour suivant. De plus, le joueur ne devra plus perdre de point lorsqu'il propose à nouveau cette lettre.
- Le programme n'affiche plus seulement le nombre de proposition. Le programme affichera au fur et à mesure un dessin qui illustrera le pendu. Une fois le dessin fini le jeu est terminé, le joueur a perdu.



- Le nombre de propositions n'est plus fixe, la longueur du mot nous donne le nombre de propositions disponible. Le nombre de propositions sera égal à sa longueur + 3.
Par exemple, un mot de 7 lettres, nous donnera $7 + 3 = 10$ propositions.
L'affichage du dessin du pendu devra s'adapter en conséquence.