

INSA

INSTITUT NATIONAL
DES SCIENCES
APPLIQUÉES
LYON

Initiation à la programmation

Enseignement du code à l'école

Séance 3

Bilan de la séance précédente

Consolidation des acquis

- Qu'avons-nous fait la dernière fois ?
 - Programmer un vaisseau pour sortir d'un labyrinthe

- Quel était notre langage de programmation ?



- Quelle méthode de travail avons-nous suivie ?
 - 1) Je réfléchis (algorithme)
 - 2) Je décris ma solution (programme)
 - 3) Je vérifie ma solution (exécution)

Glossaire ...

- Algorithme :
- Programme :
- Instruction :
- Bug :
- Déplacements relatifs :

- Algorithme : *description **générale** d'une solution à un problème*
- Programme : ***instructions** qui respectent un algorithme*
- Instruction : *action exécutable par **l'ordinateur***
- Bug : *erreur dans un programme*
- Déplacements relatifs : *consignes données par rapport à l'objet*

Culture générale : révisions

Qui sont ils ?



Comment sont ils connectés ?



Écran



Souris

Clavier



Unité centrale

Comment sont ils connectés ?



Écran



HDMI



Clavier



USB



Souris



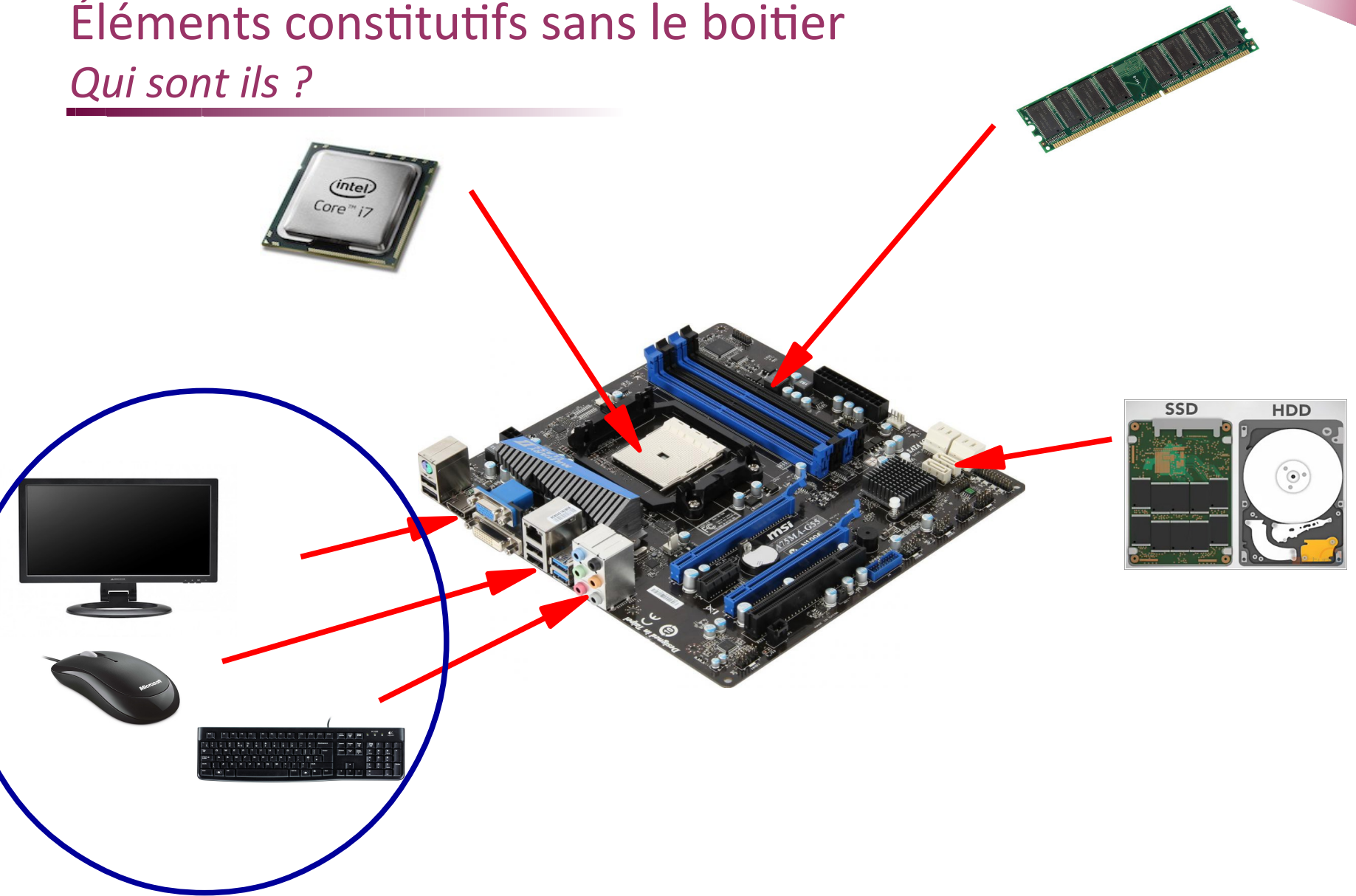
USB



Unité centrale

Éléments constitutifs sans le boîtier

Qui sont ils ?

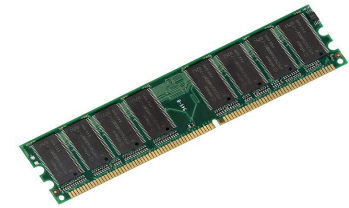


Éléments constitutifs sans le boîtier

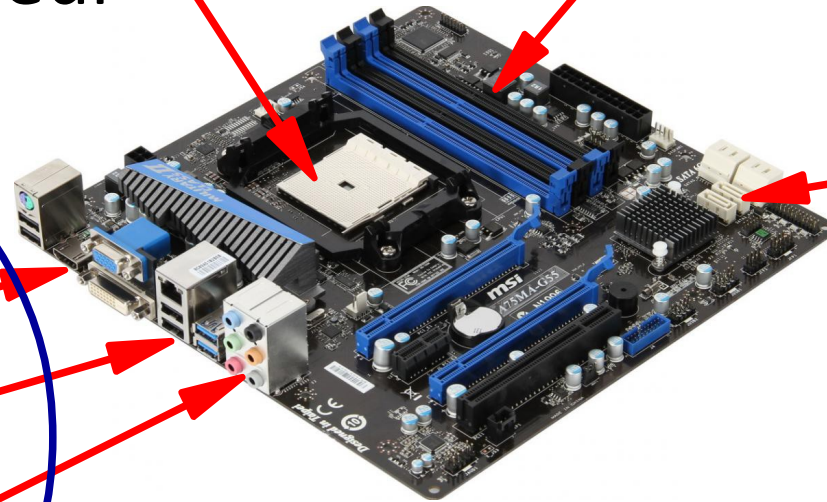
Qui sont ils ?



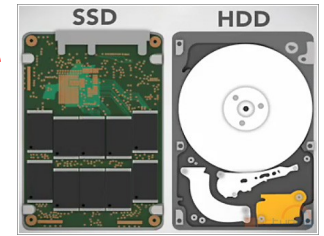
Processeur



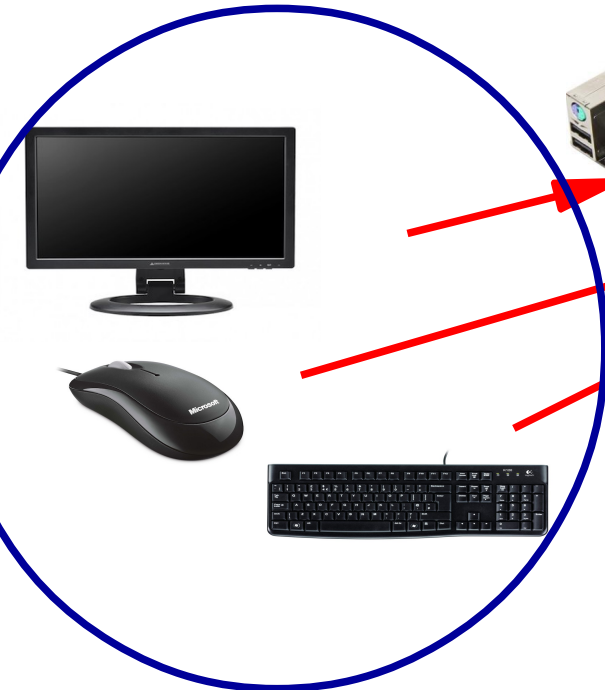
Mémoire vive
(RAM)



Carte mère



Mémoire de
masse
(Disque dur)



Périphériques

Lancement de la séance 3

Boucles d'exécution

Définition du langage de programmation du jour

- Programmation de déplacements **relatifs**

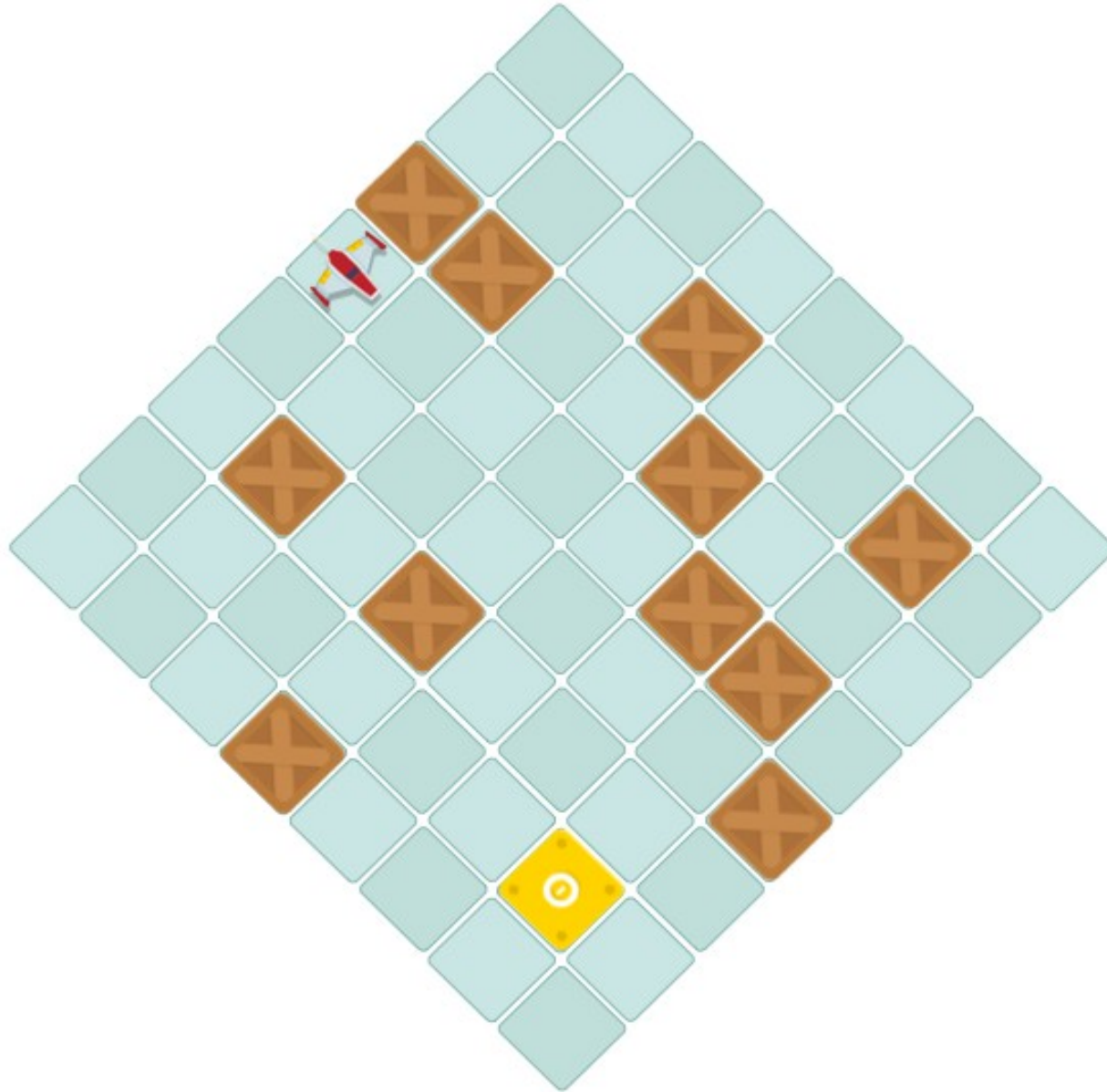
Centrés sur le vaisseau (autocentrés)

- Possibilité de répéter des actions

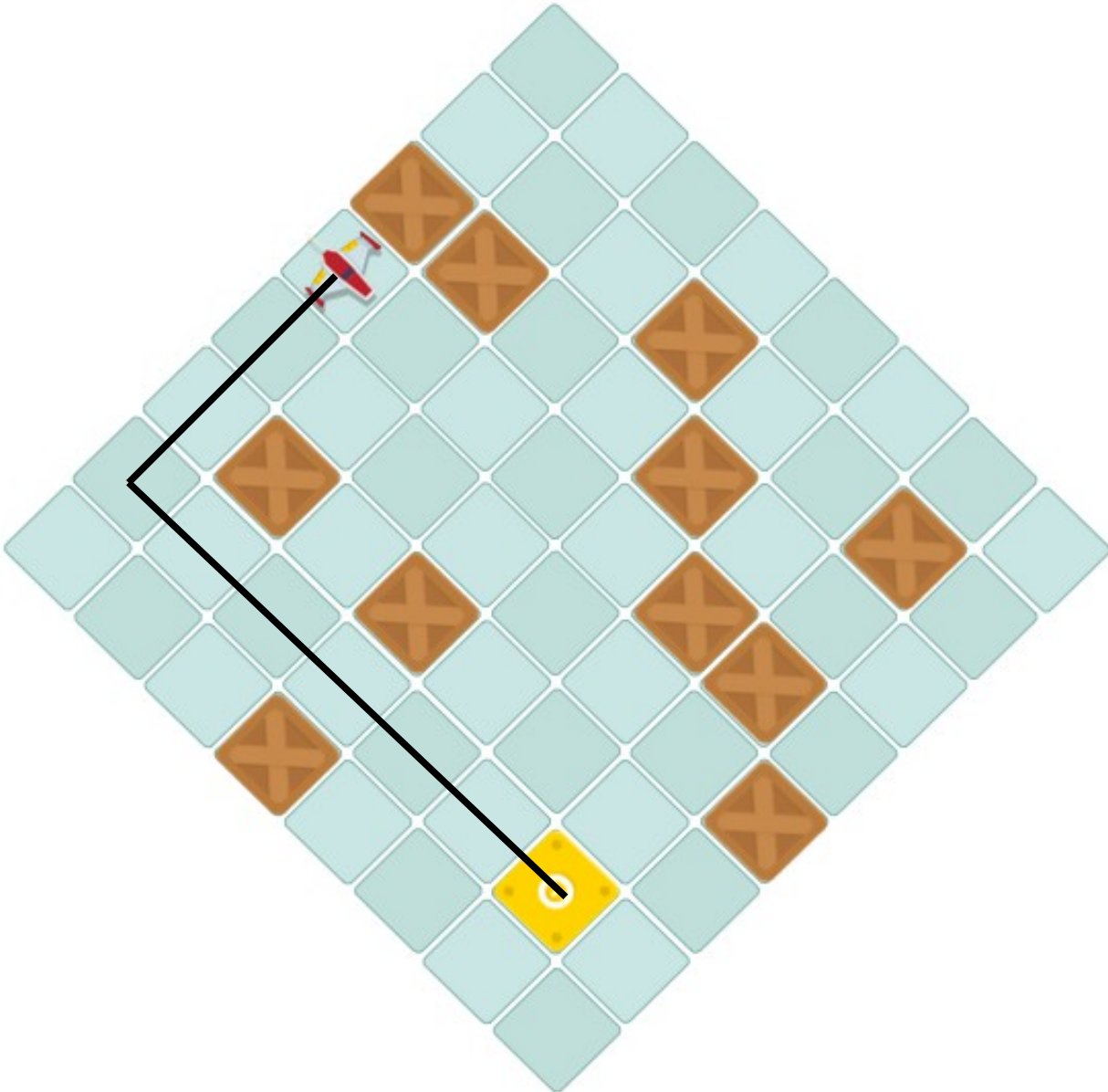
- Proposition d'un langage de programmation :



Exemple

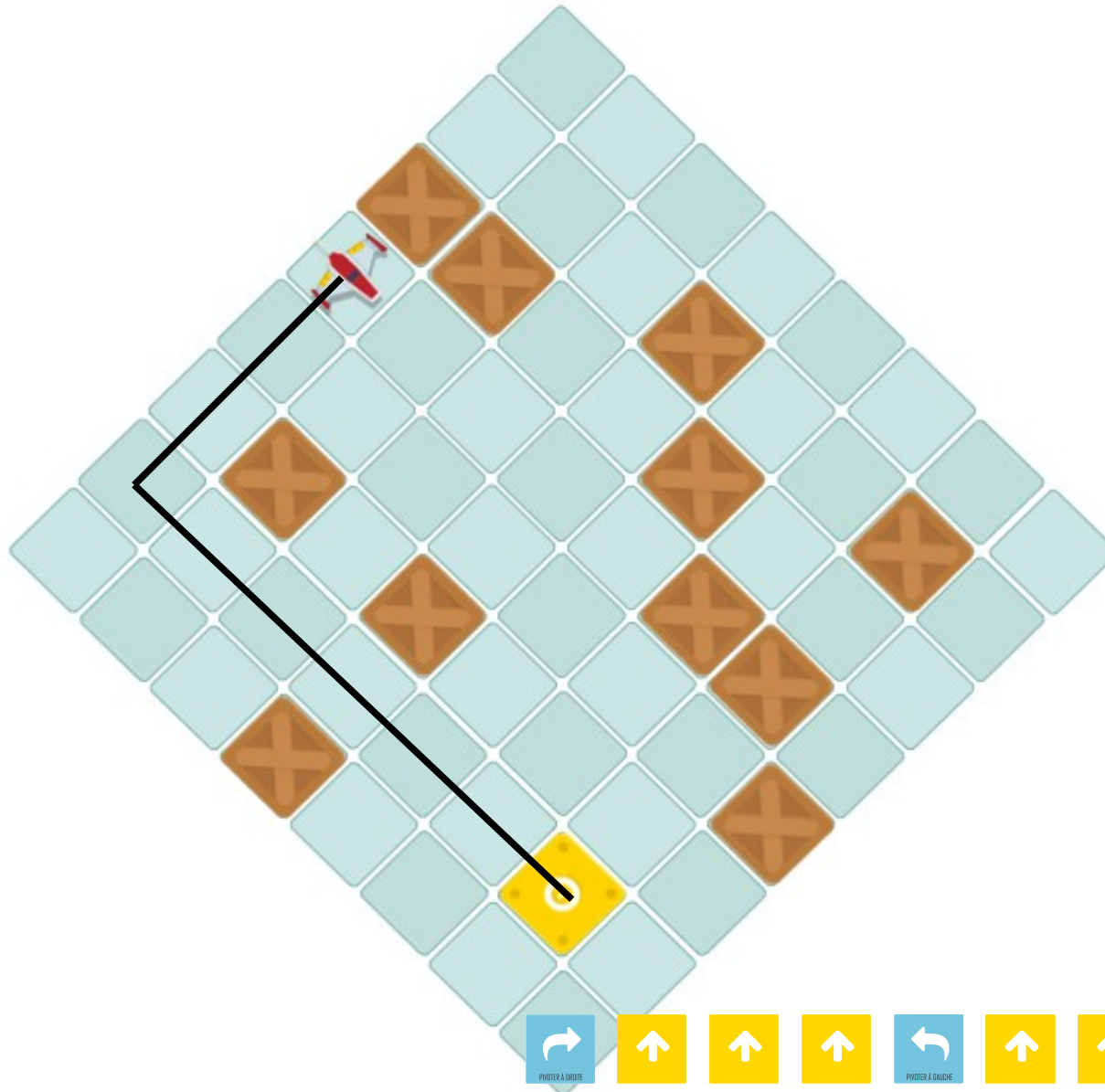


Exemple



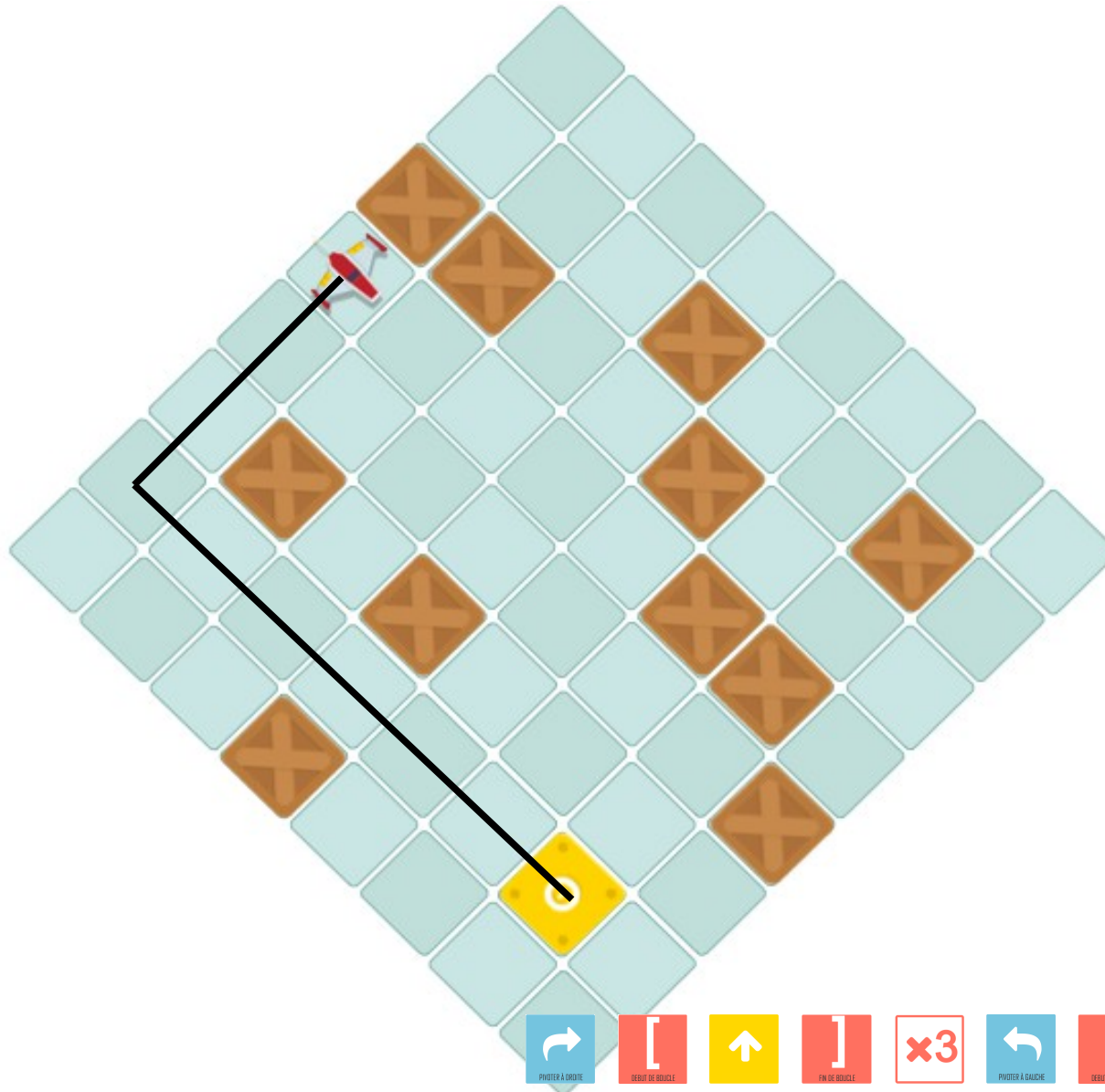
Exemple

Solution 1 (sans boucle) : 11 instructions



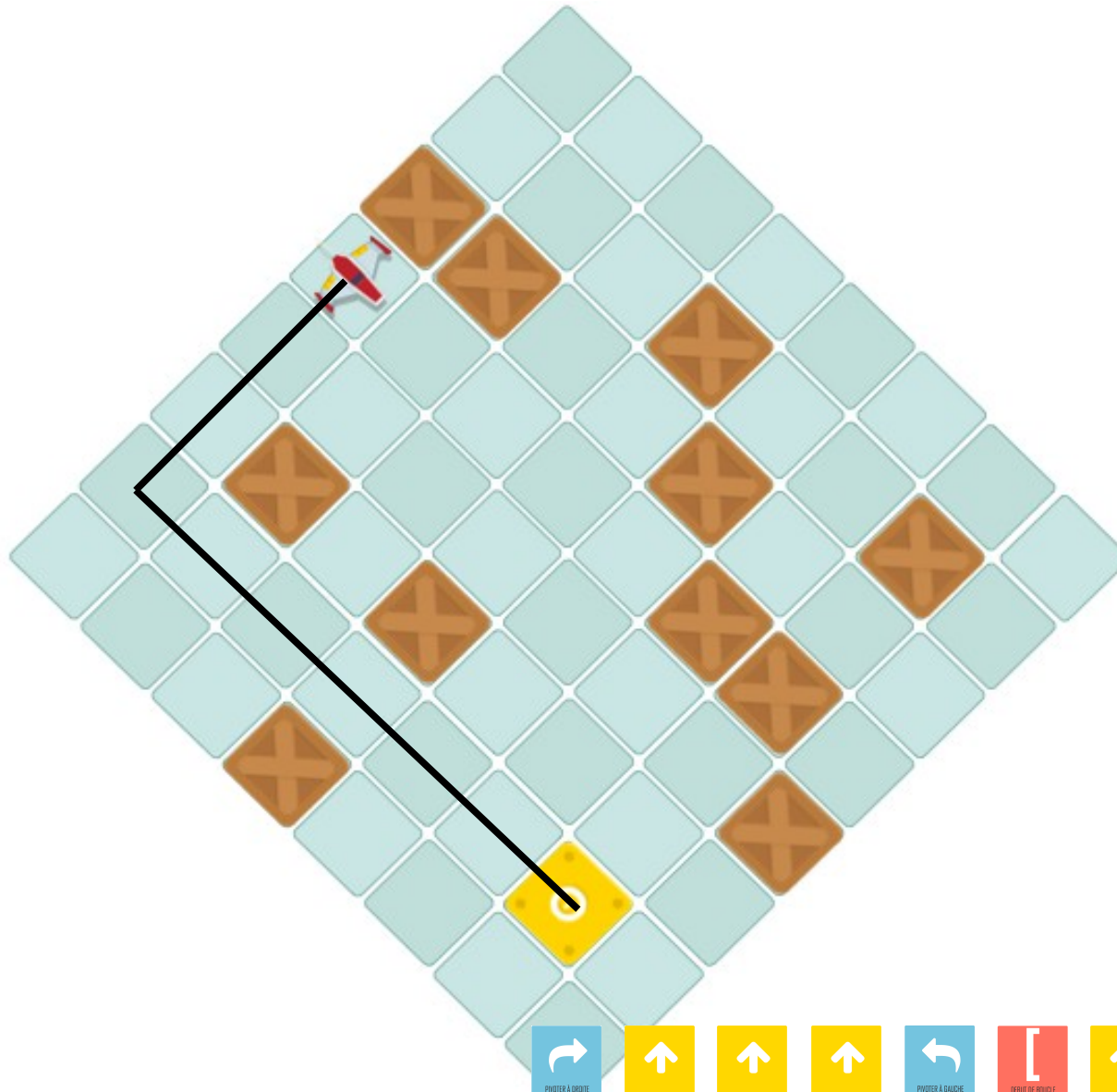
Exemple

Solution 2 (Que des boucles) : 10 instructions



Exemple

Solution 3 (boucles quand c'est mieux) : 9 instructions



Méthode de développement à suivre

1) Ensemble :

- Dessiner l'algorithme

2) Seul :

- Écrire un programme correspondant
- **Vérifier** le programme écrit

3) Ensemble :

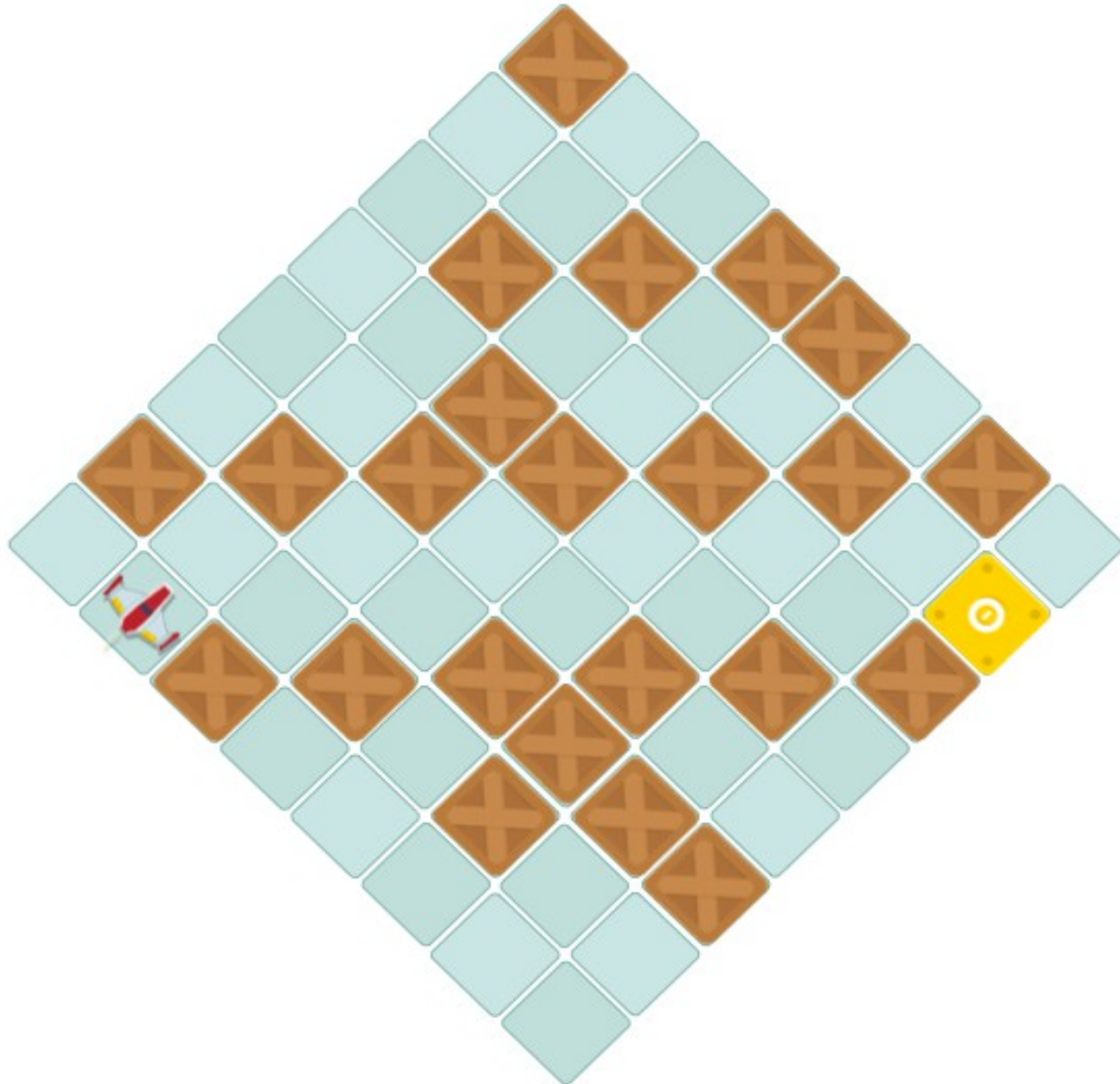
- Choisir 1 programme
- **Vérifier** le programme et **marquer son nom** !

Une solution **correcte** apporte +1 point d'équipe !

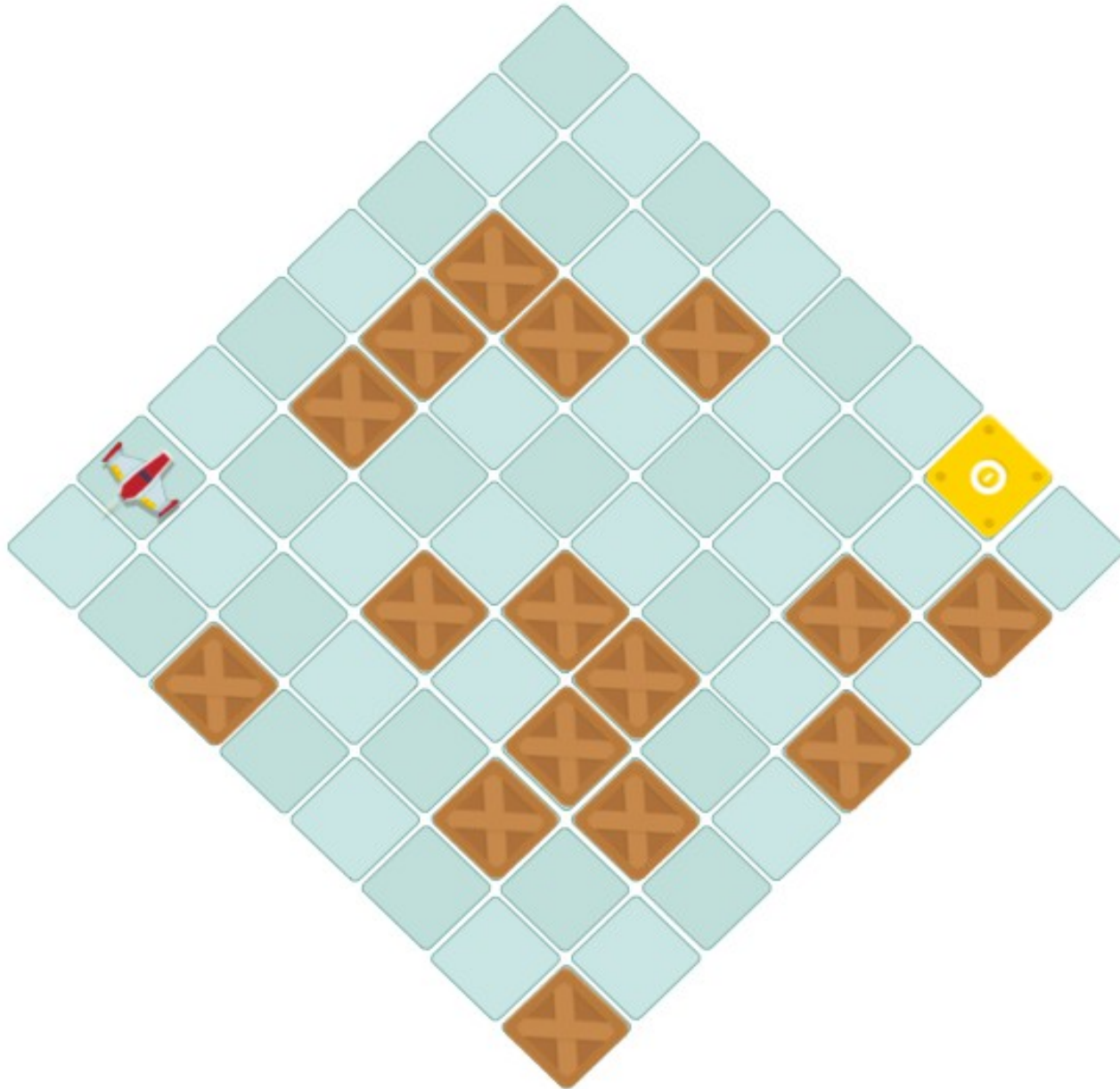
La solution correcte la plus **courte** rapporte +1 point d'équipe !

La solution correcte la plus **rapide** rapporte +1 point d'équipe !

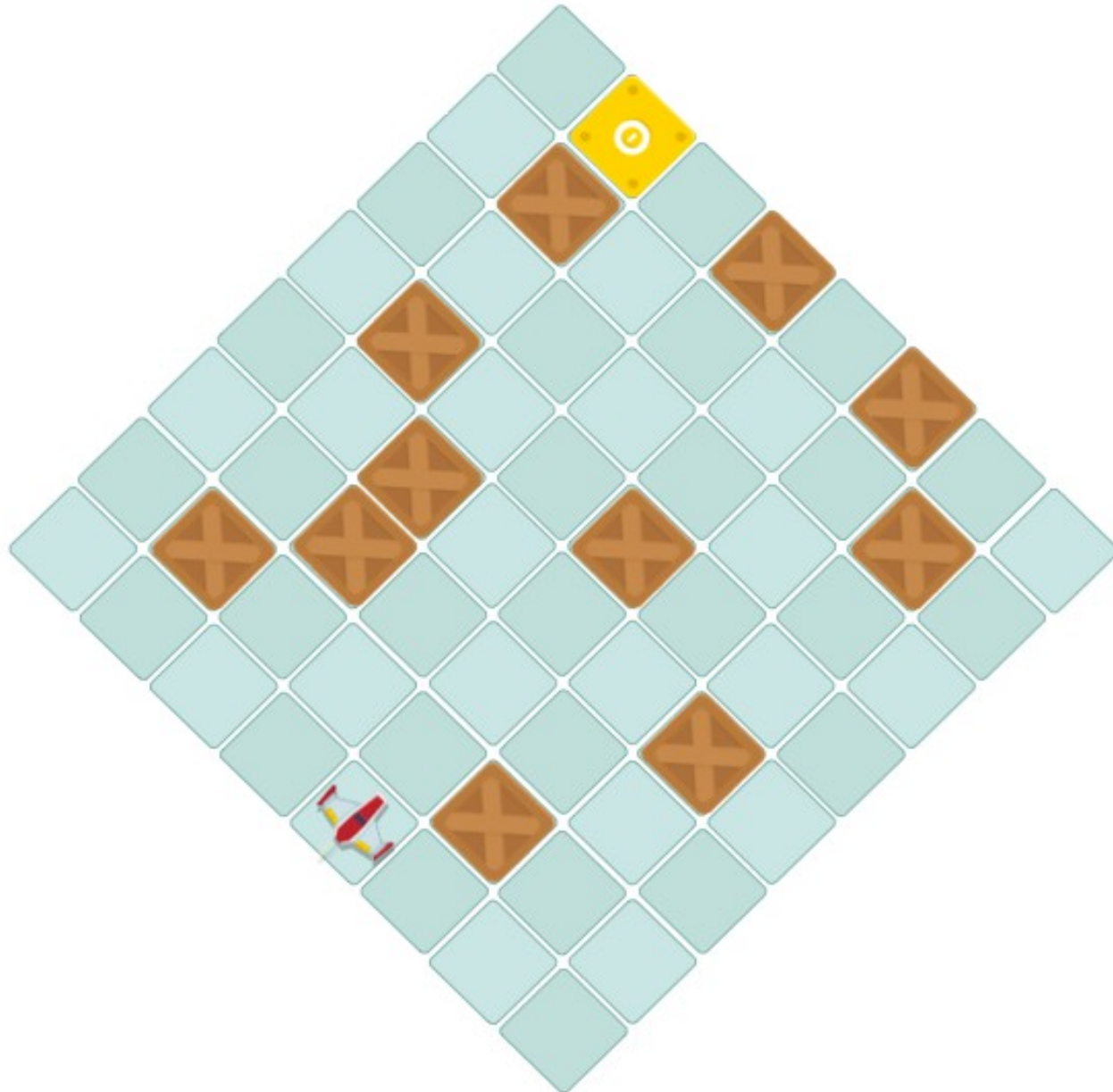
À vous de travailler jouer ! (Fiche 9)



À vous de travailler jouer ! (Fiche 10)



À vous de travailler jouer ! (Fiche 12)



Bilan

Qu'avons nous vu durant ces 3 séances ?

■ Deux **langages** de programmation

– Déplacements absolus    

– Déplacements relatifs avec boucles      

■ Une **méthode** de travail

1) Je réfléchis (algorithme)

2) Je décris ma solution (programme)

3) Je vérifie ma solution (exécution)

■ Du **vocabulaire**

Algorithme, programme, instruction, bug, périphérique, processeur, mémoire vive, mémoire de masse, ...

■ « Initiation à l'informatique »

- Culture générale numérique
- Spatialisation
- Initiation à la pensée algorithmique

■ *Pour aller plus loin :*

- *TuxBot* <http://appli-etna.ac-nantes.fr:8080/ia53/tice/ressources/tuxbot/index.php>
- *Studio code* <https://studio.code.org/hoc/1>
- *Scratch* <https://scratch.mit.edu/>
- *Jeux programmables par blocs* <https://blockly-games.appspot.com/>
- « *J'apprends à programmer tout seul !* » ISBN 2047353998
- « *1, 2, 3... codez !!* » ISBN 978-2-7465-1106-4
- *Jack la souris robot*
- *Beebot*

Merci à vous pour votre accueil !

Bonne continuation et RDV dans 11 ans, à l'INSA Lyon