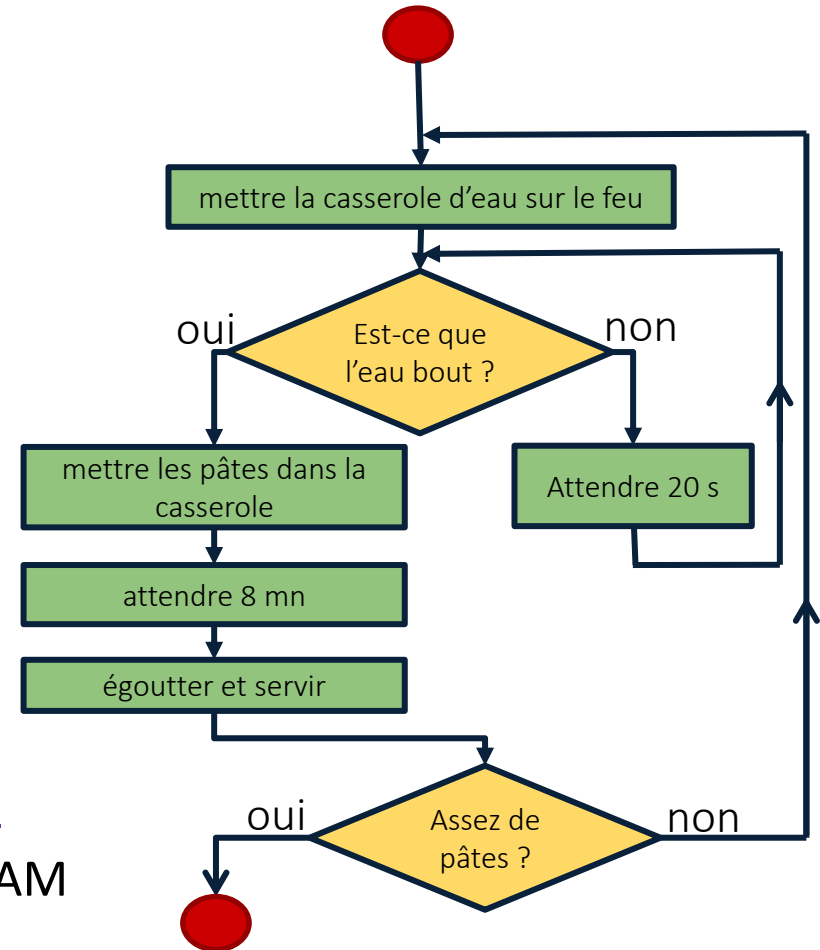


Qu'est-ce qu'un Algorithme ?

Pierre Huguet

Présentation inspirée de celle du [Pr Morelli](#) [mobile CSP](#)
et de travaux conduits avec [Marie Caroline Gries](#) au CNAM



PLAN

<https://onvaessayer.org?playlist=algorithmes>

1. Qu'est-ce qu'un algorithme

<https://onvaessayer.org?video=algoDefinition>

- Définition
- Caractéristiques et structure d'un algorithme
 - Précis (sans ambiguïté) avec des instructions réalisables
 - Structure : séquences d'instructions, répétitions, sélections/conditions

2. Écrire ou décrire un algorithme :

<https://onvaessayer.org?video=algoEcrire>

- du langage naturel au langage de programmation
- Le pseudo code et les diagrammes de flux (flowchart)

3. Exercices sur pseudo-code et diagrammes

<https://onvaessayer.org?video=algoExercices>

- Énoncés et corrigés

<https://onvaessayer.org?video=algoSolutions>

4. Variables et procédures

<https://onvaessayer.org?video=algoVariablesEtProcedures>

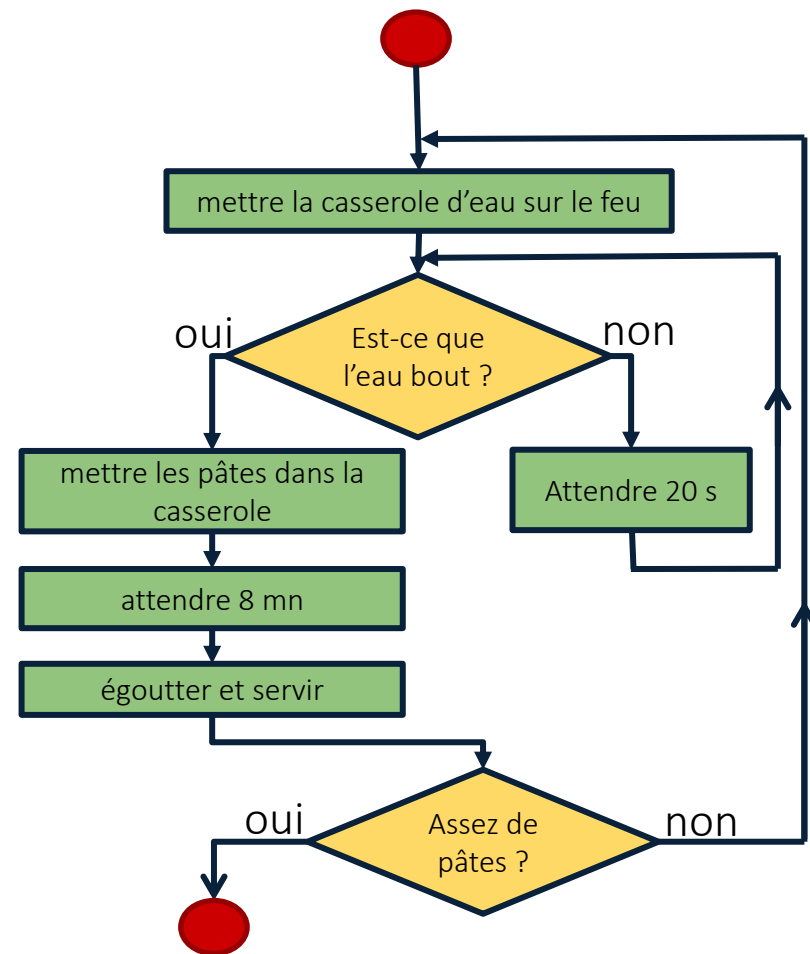
voir ensuite ... le "computational thinking"

<https://onvaessayer.org?video=computationalThinking>

1a

qu'est-ce
qu'un
algorithme ?

Pierre Huguet



QU'EST-CE QU'UN ALGORITHME ?

- Définition du Larousse
- *Un **algorithme** est un ensemble de **règles opératoires** dont l'application permet de **résoudre un problème** énoncé, au moyen d'un nombre fini d'opérations.*

QU'EST-CE QU'UN ALGORITHME ?

- Définition du Larousse
- *Un **algorithme** est un ensemble de **règles opératoires** dont l'application permet de **résoudre un problème** énoncé, au moyen d'un nombre fini d'opérations. Un **algorithme** peut être traduit, grâce à un langage de programmation, **en programme** exécutable par un ordinateur.*

QU'EST-CE QU'UN ALGORITHME ?

- Définition du Larousse

- Un *algorithme* est un ensemble de *règles opératoires* dont l'application permet de *résoudre un problème* énoncé, au moyen d'un nombre fini d'opérations. Un *algorithme peut être traduit*, grâce à un langage de programmation, *en programme* exécutable par un ordinateur.

- Définition Wikipedia

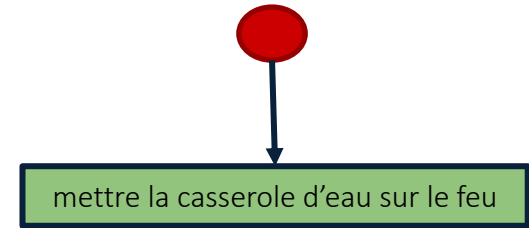
- Un *algorithme* est une *suite finie et non ambiguë* d'opérations ou d'instructions permettant de *résoudre un problème* ou d'obtenir un *résultat*

QU'EST-CE QU'UN ALGORITHME ?

- Un *algorithme* est similaire à une *recette*.

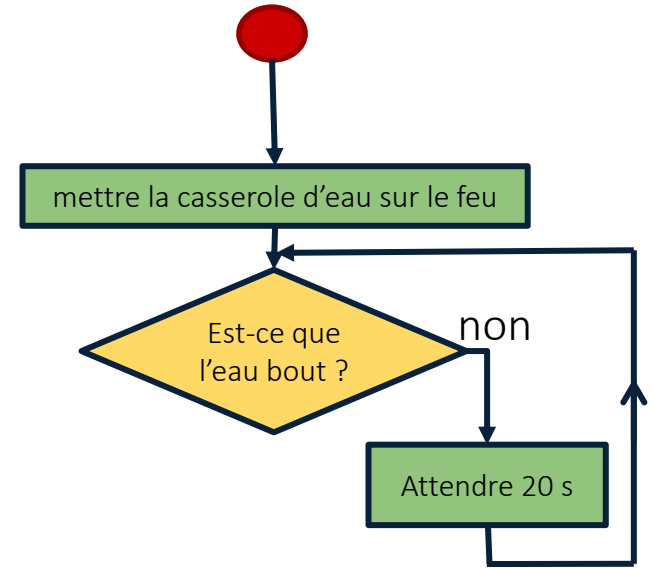
QU'EST-CE QU'UN ALGORITHME ?

- Un *algorithme* est similaire à une *recette*.
- Pour cuire des pâtes :
 - mettre la casserole sur le feu.



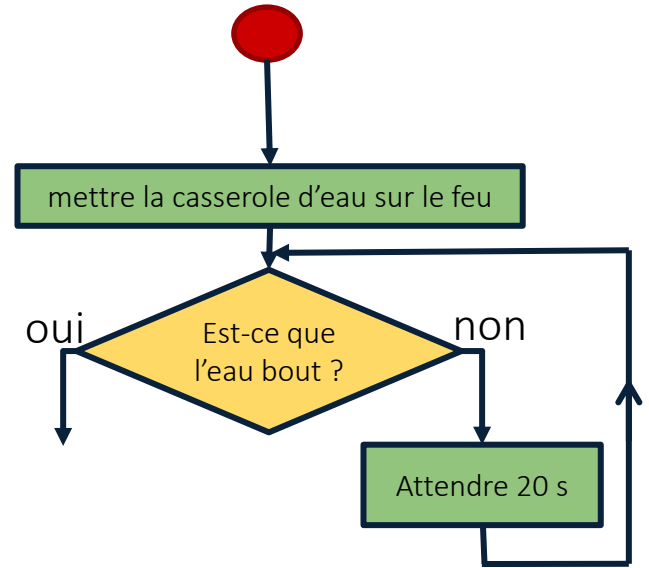
QU'EST-CE QU'UN ALGORITHME ?

- Un *algorithme* est similaire à une *recette*.
- Pour cuire des pâtes :
 - mettre la casserole sur le feu.
 - est-ce que l'eau bout ?
 - Si non : attendre un peu et vérifier à nouveau



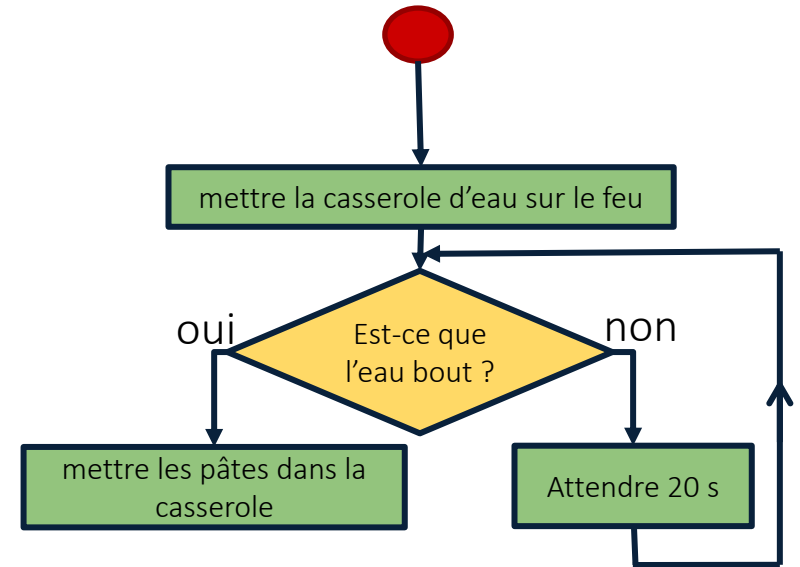
QU'EST-CE QU'UN ALGORITHME ?

- Un *algorithme* est similaire à une *recette*.
- Pour cuire des pâtes :
 - mettre la casserole sur le feu.
 - est-ce que l'eau bout ?
 - Si non : attendre un peu et vérifier à nouveau
 - Si oui : continuer



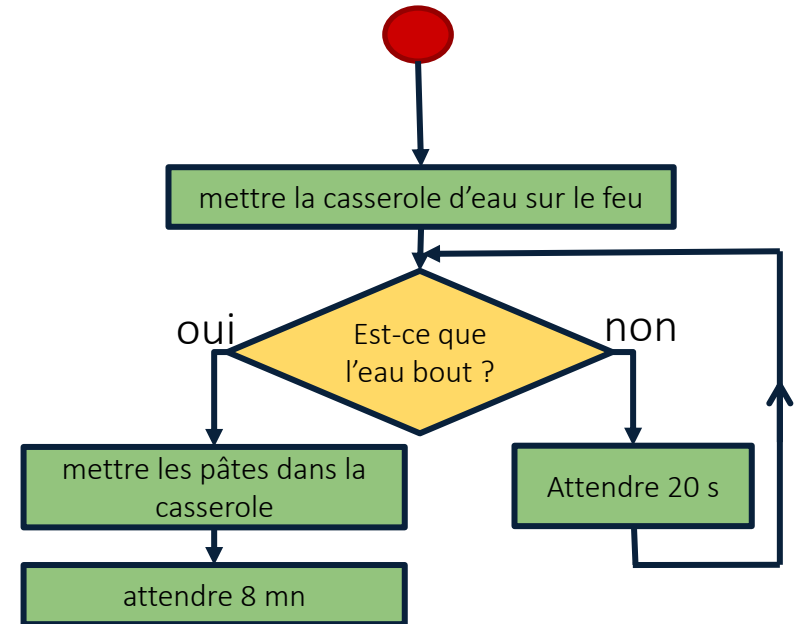
QU'EST-CE QU'UN ALGORITHME ?

- Un *algorithme* est similaire à une *recette*.
- Pour cuire des pâtes :
 - mettre la casserole sur le feu.
 - est-ce que l'eau bout ?
 - Si non : attendre un peu et vérifier à nouveau
 - Si oui : continuer
 - mettre les pâtes dans la casserole



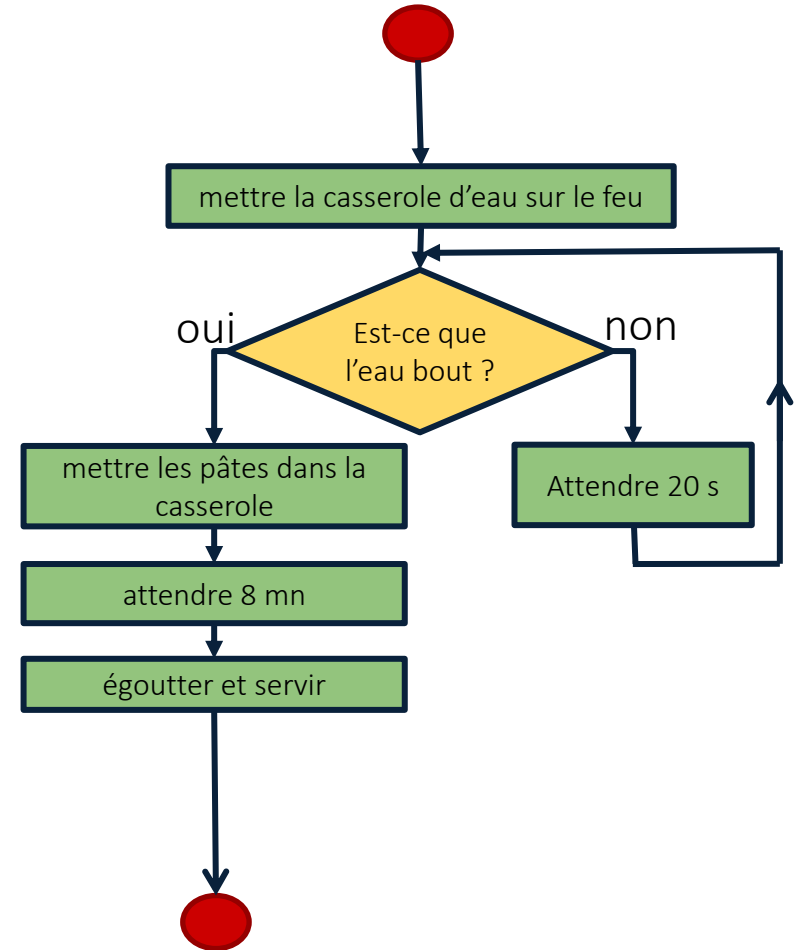
QU'EST-CE QU'UN ALGORITHME ?

- Un *algorithme* est similaire à une *recette*.
- Pour cuire des pâtes :
 - mettre la casserole sur le feu.
 - est-ce que l'eau bout ?
 - Si non : attendre un peu et vérifier à nouveau
 - Si oui : continuer
 - mettre les pâtes dans la casserole
 - attendre 8 mn



QU'EST-CE QU'UN ALGORITHME ?

- Un *algorithme* est similaire à une *recette*.
- Pour cuire des pâtes :
 - mettre la casserole sur le feu.
 - est-ce que l'eau bout ?
 - Si non : attendre un peu et vérifier à nouveau
 - Si oui : continuer
 - mettre les pâtes dans la casserole
 - attendre 8 mn
 - égoutter et servir

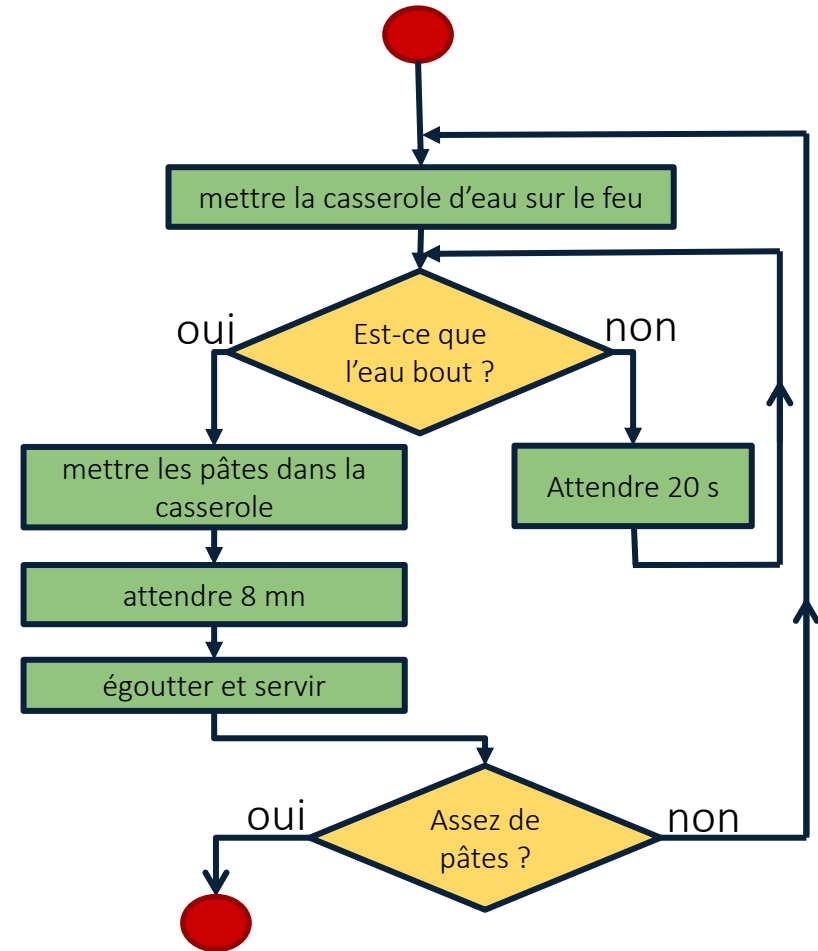


1. QU'EST-CE QU'UN ALGORITHME ?

Un *algorithme* est similaire à une *recette*.

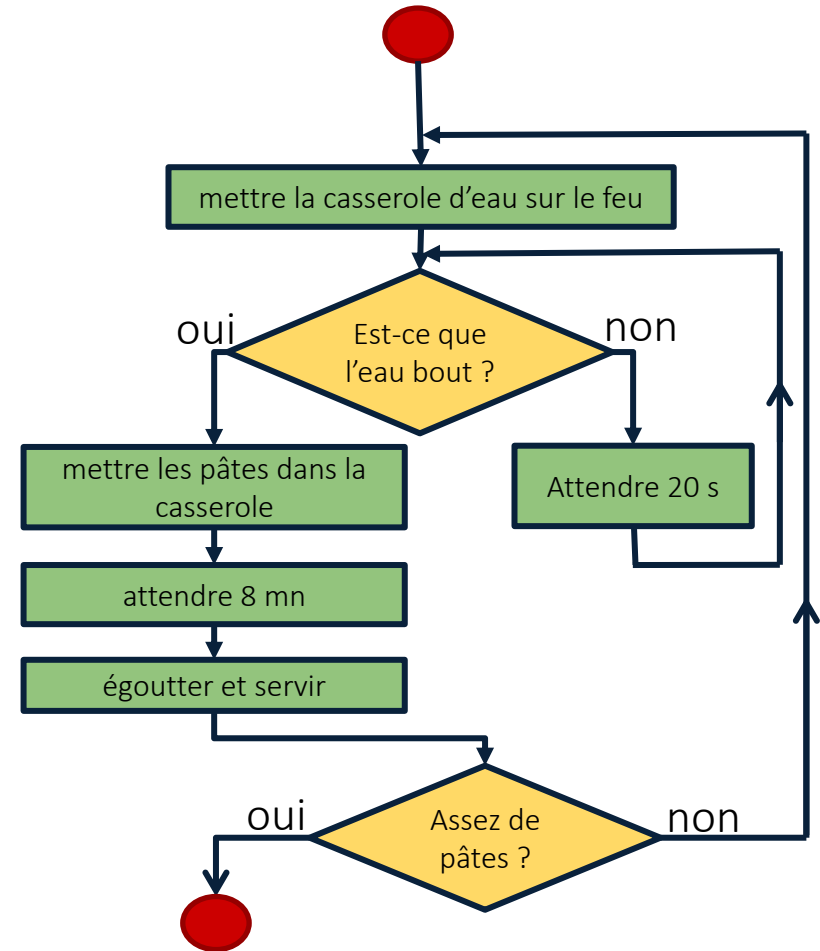
Pour cuire des pâtes :

- mettre la casserole sur le feu.
- est-ce que l'eau bout ?
 - Si non : attendre un peu et vérifier à nouveau
 - Si oui : continuer
- mettre les pâtes dans la casserole
- attendre 8 mn
- égoutter et servir
- répéter s'il n'y a pas assez de pâtes



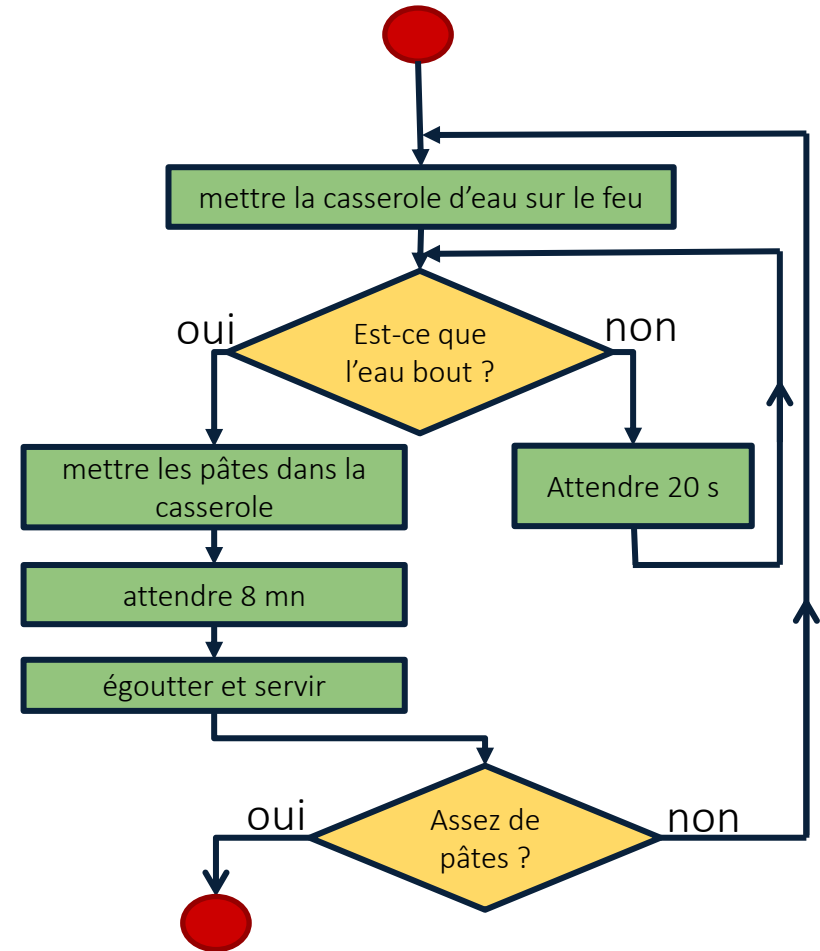
QU'EST-CE QU'UN ALGORITHME ?

- Un *algorithme* est similaire à une *recette*.
- Pour cuire des pâtes :
 - mettre la casserole sur le feu.
 - est-ce que l'eau bout ?
 - Si non : attendre un peu et vérifier à nouveau
 - Si oui : continuer
 - mettre les pâtes dans la casserole
 - attendre 8 mn
 - égoutter et servir
 - répéter s'il n'y a pas assez de pâtes
- Différence entre *algorithme* et *recette* ?



QU'EST-CE QU'UN ALGORITHME ?

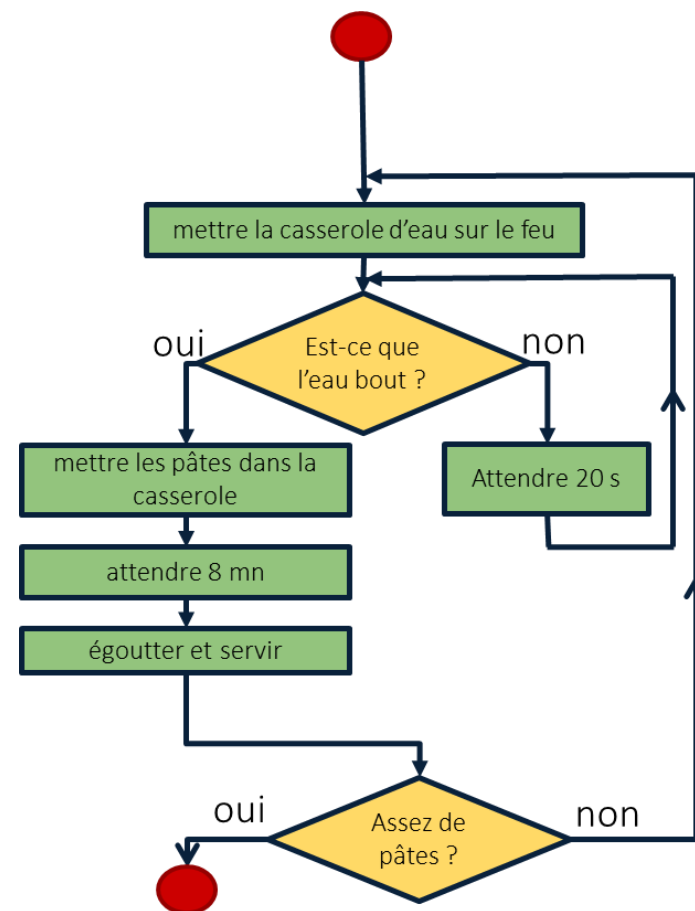
- Un *algorithme* est similaire à une *recette*.
- Pour cuire des pâtes :
 - mettre la casserole sur le feu.
 - est-ce que l'eau bout ?
 - Si non : attendre un peu et vérifier à nouveau
 - Si oui : continuer
 - mettre les pâtes dans la casserole
 - attendre 8 mn
 - égoutter et servir
 - répéter s'il n'y a pas assez de pâtes
- Différence entre *algorithme* et *recette* ?
- Les algorithmes informatiques sont plus précis.



1b

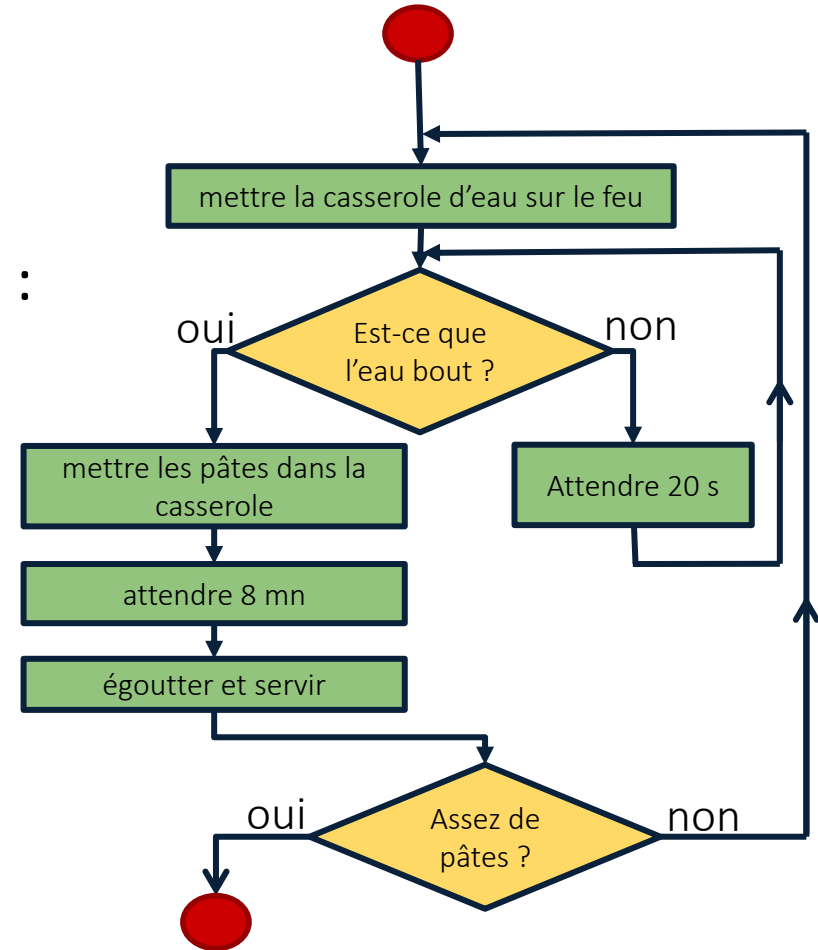
Caractéristiques et structure d'un algorithme

Pierre Huguet



CARACTÉRISTIQUES ET STRUCTURE D'UN ALGORITHME

- Un algorithme doit être précis.
- Un algorithme est défini pas-à pas (séquence) :
 - Chaque pas ou étape doit être précis et faisable.
- Tous les algorithmes peuvent être réalisés à partir de 3 "*structures de contrôle*"
 - séquences : suite ordonnée d'instructions
 - sélections : tests, si ... sinon ...
 - répétitions : boucles



UN ALGORITHME DOIT ÊTRE PRÉCIS

- Attention à l'ambiguïté :
 - *“Chéri, va au supermarché, prends une boîte de lentilles, si ils ont des œufs, prends en douze.”*

UN ALGORITHME DOIT ÊTRE PRÉCIS

- Attention à l'ambiguïté :
 - . *“Chéri, va au supermarché, prends une boîte de lentilles, si ils ont des œufs, prends en douze.”*
 - . *et Chéri revient avec ...*

UN ALGORITHME DOIT ÊTRE PRÉCIS

- Attention à l'ambiguïté :
 - *“Chéri, va au supermarché, prends une boîte de lentilles, si ils ont des œufs, prends en douze.”*
 - *et Chéri revient avec 12 boîtes de lentilles.*

UN ALGORITHME DOIT ÊTRE PRÉCIS

- Attention à l'ambiguïté :
 - . *“Chéri, va au supermarché, prends une boîte de lentilles, si ils ont des œufs, prends en douze.”*
 - . *et Chéri revient avec 12 boîtes de lentilles.*
 - . *Problème ?*
 - Chéri a fait ce qui lui a été demandé ...*
 - mais une douzaine de quoi ?*
 - l'algorithme est ambigu, non ?*

UN ALGORITHME DOIT ÊTRE PRÉCIS

- Attention à l'ambiguïté :
 - . *“Chéri, va chercher une boîte de lentilles au supermarché et tant que tu y es prends des œufs.”*
 - . *et Chéri ...*

UN ALGORITHME DOIT ÊTRE PRÉCIS

- Attention à l'ambiguïté :
 - . *“Chéri, va chercher une boîte de lentilles au supermarché et tant que tu y es prends des œufs.”*
 - . *et Chéri n'est jamais revenu ...*

UN ALGORITHME DOIT ÊTRE PRÉCIS

- Attention à l'ambiguïté :
 - . *“Chéri, va chercher une boîte de lentilles au supermarché et tant que tu y es prends des oeufs.”*
 - . *et Chéri n'est jamais revenu ...*
- *Problème ?*
 - Chéri a fait ce qui lui a été demandé ...*
 - "tant qu'il y était" il a continué à prendre des œufs ...*
 - Boucle infinie ...*

POINTS CLEFS SUR LES ALGORITHMES

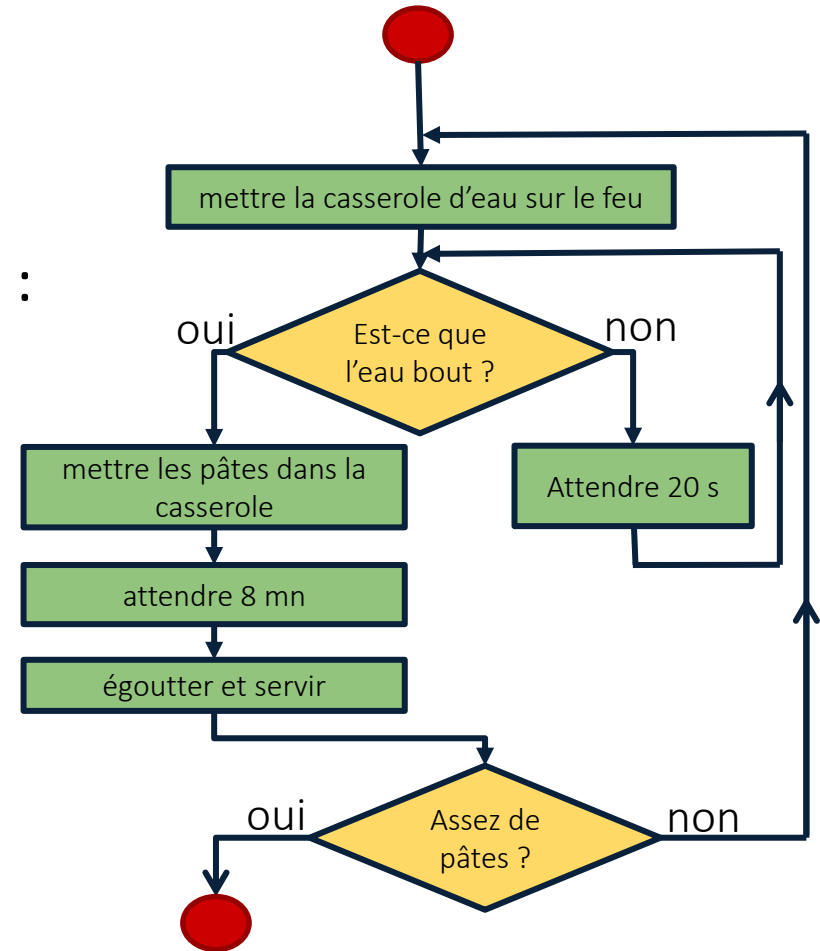
Un algorithme doit être précis.

Un algorithme est défini pas-à pas (séquence) :

Chaque pas ou étape doit être précis et faisable.

POINTS CLEFS SUR LES ALGORITHMES

- Un algorithme doit être précis.
- Un algorithme est défini pas-à pas (séquence) :
 - Chaque pas ou étape doit être précis et faisable.
- Tous les algorithmes peuvent être réalisés à partir de
 - séquences : suite ordonnée d'instructions
 - sélections : tests, si ... sinon ...
 - répétitions : boucles



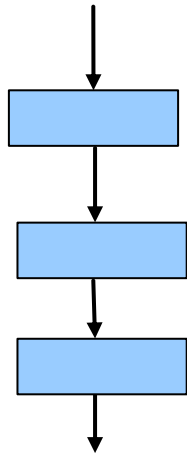
BASE DES ALGORITHMES : STRUCTURES DE CONTRÔLE

la séquence,

la sélection,

et

la *répétition*.



Séquence

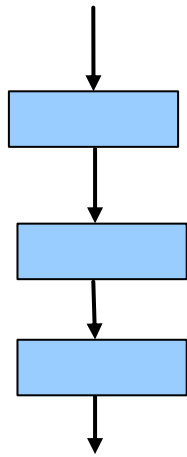
BASE DES ALGORITHMES : STRUCTURES DE CONTRÔLE

la séquence,

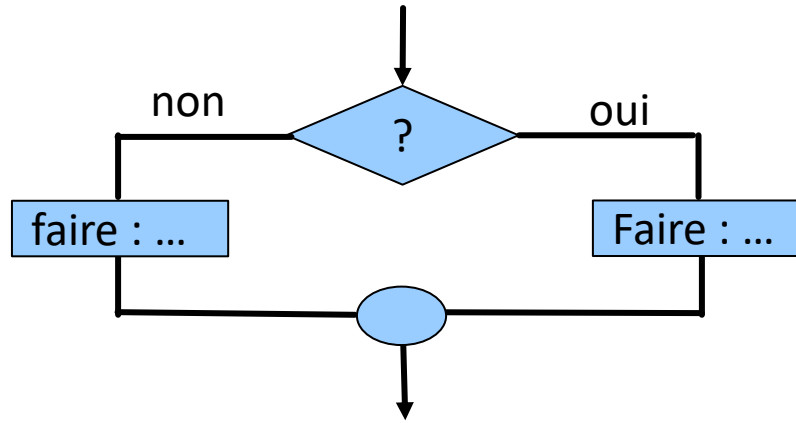
la sélection,

et

la *répétition*.



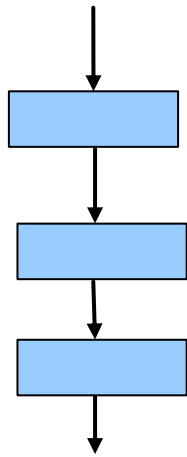
Séquence



Sélection
ou condition
(Branchement conditionnel)

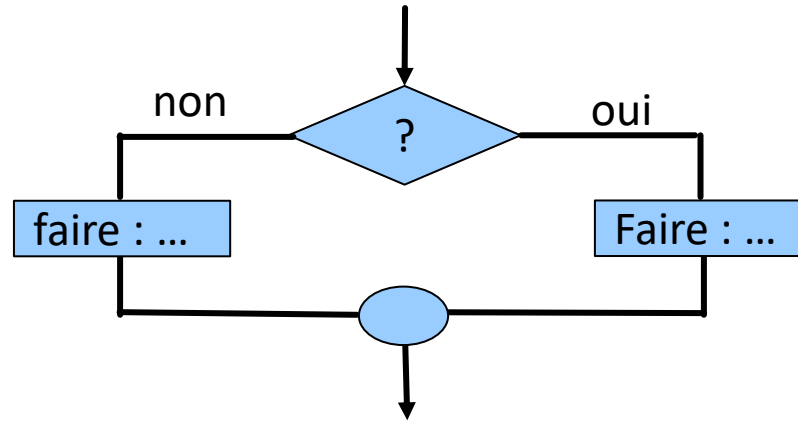
BASE DES ALGORITHMES : STRUCTURES DE CONTRÔLE

la séquence,



Séquence

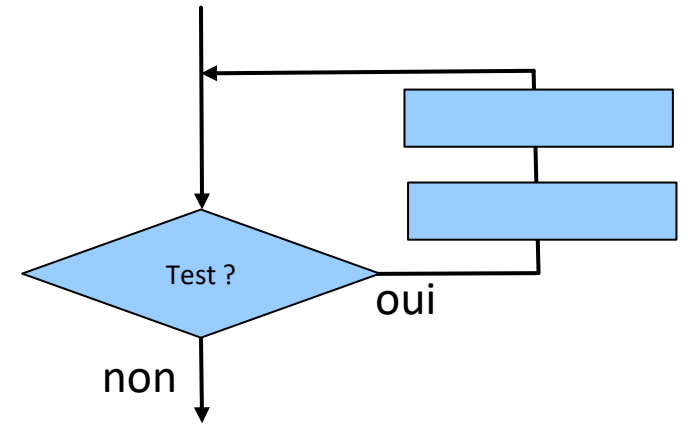
la sélection,



Sélection
ou condition
(Branchement conditionnel)

et

la *répétition*.



Répétition
ou itération
(Boucle)

1° STRUCTURE DE CONTRÔLE : LA SÉQUENCE

Une *Séquence* est une suite d'instructions que l'ordinateur exécute une par une, dans l'ordre où elles sont écrites

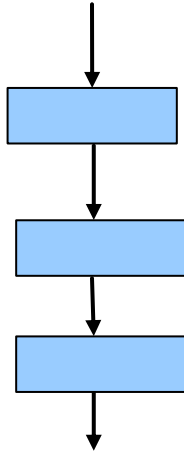


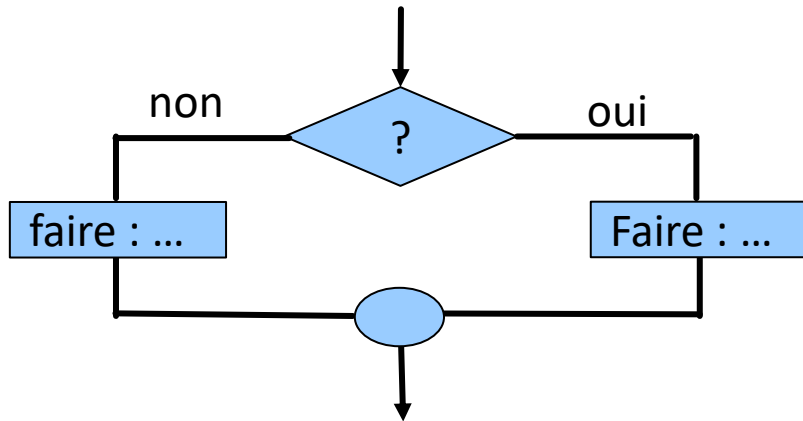
Diagramme d'une Séquence –
chaque rectangle correspond à une instruction.

Algorithme de calcul de la moyenne de 70, 80, et 90.



2° STRUCTURE DE CONTRÔLE : LA SÉLECTION (OU CONDITION)

La *Sélection* utilise une condition booléenne (vraie ou fausse) pour décider de la suite ou de la branche des instructions à exécuter.



Algorithme pour décider si "note" est une note suffisante.

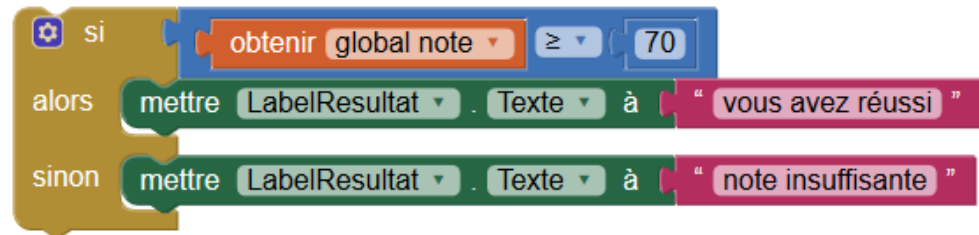
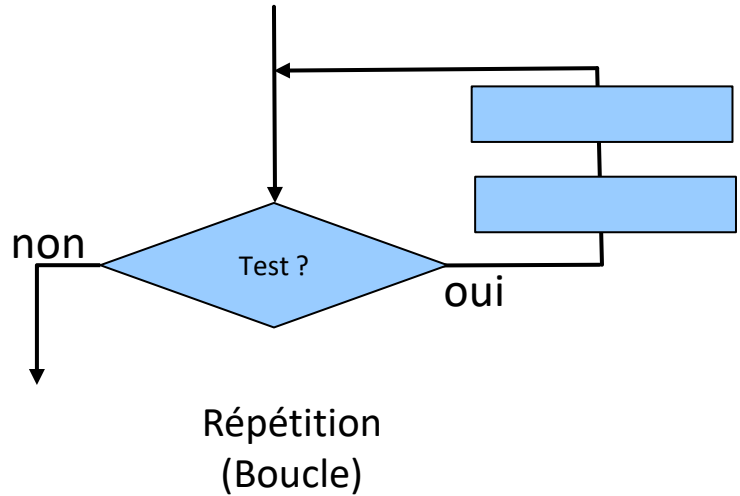


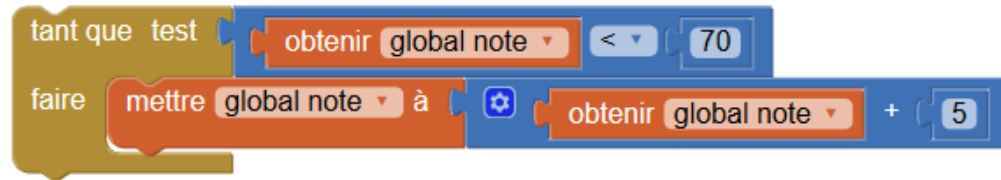
Diagramme de sélection -- le losange exprime une condition; le cercle un connecteur.

3° STRUCTURE DE CONTRÔLE : LA RÉPÉTITION (BOUCLE OU ITÉRATION)

La *répétition* (ou *itération*) c'est la répétition de la partie d'un algorithme. On peut préciser le nombre de fois ou effectuer la répétition tant qu'une condition est remplie.



algorithme qui ajoute 5 points à la note jusqu'à ce qu'elle atteigne ou dépasse 70.



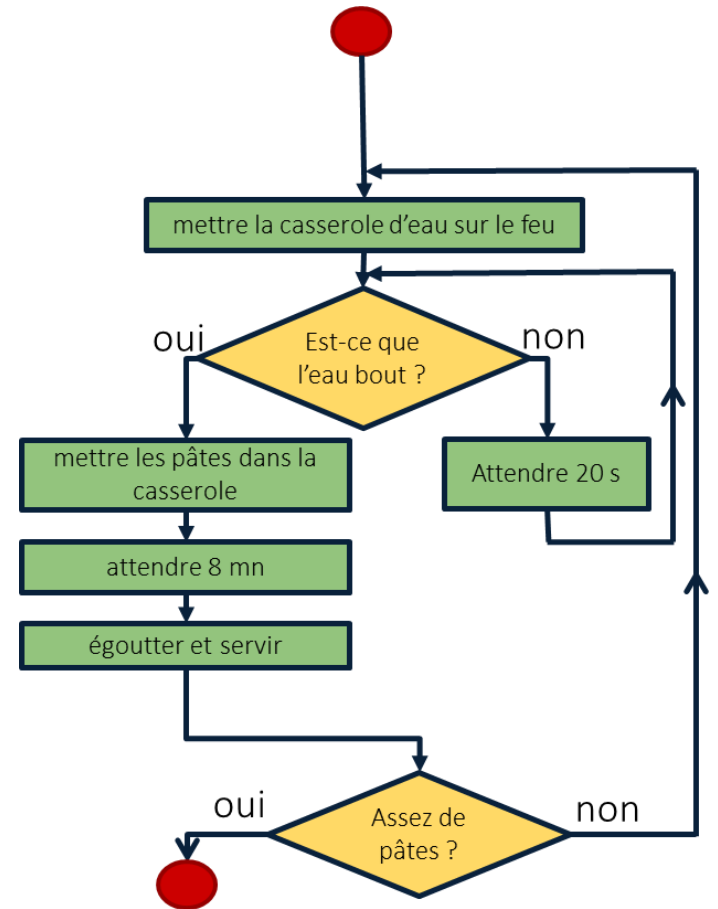
POINTS CLEFS SUR LES ALGORITHMES

- Un algorithme doit être précis.
- Un algorithme est défini pas-à pas (séquence) :
 - Chaque pas ou étape doit être précis et faisable.
- Tous les algorithmes peuvent être réalisés à partir de
 - séquences : suite ordonnée d'instructions
 - sélections : tests, si ... sinon ...
 - répétitions : boucles

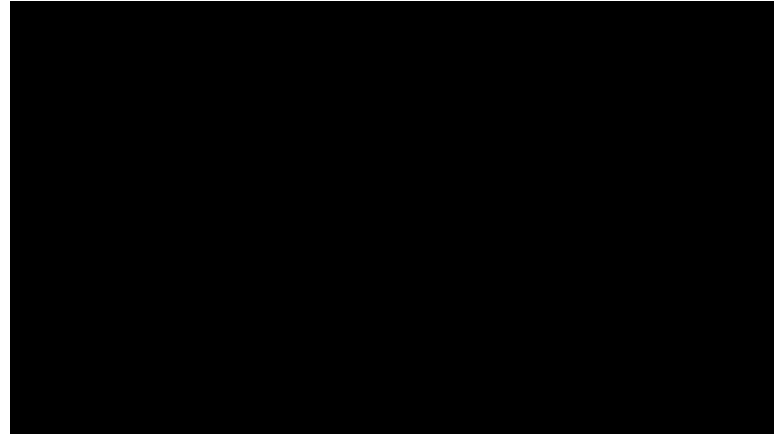
2

écrire ou décrire un algorithme

Pierre Huguet



DES ALGORITHMES TOUS LES JOURS ...

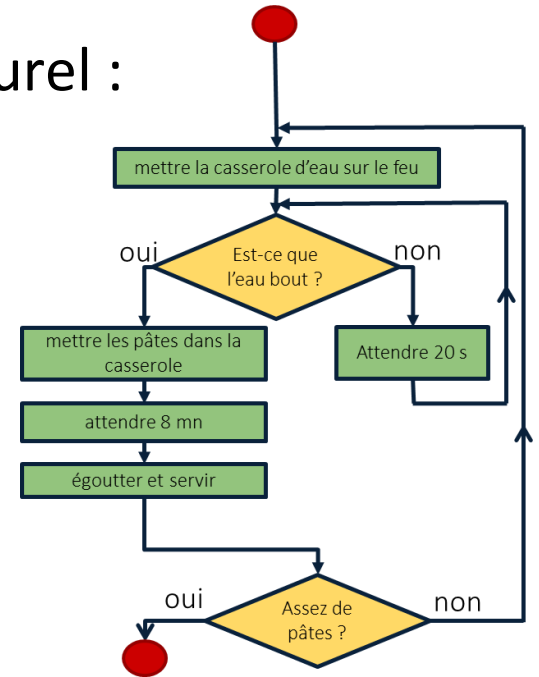


ECRIRE OU DÉCRIRE UN ALGORITHME

- Les algorithmes peuvent être exprimés en langage naturel :
 - pour faire des pâtes, mettre de l'eau dans une casserole, portez à ébullition, puis ajoutez ...
- ou par des diagrammes de flux (flowchart)
- ou par du pseudo-code (notez le décalage)
- ou avec un langage de exploitable par un ordinateur

Pseudo code

```
répéter jusqu'à "assez de pâtes est vrai"  
  mettre la casserole d'eau sur le feu  
  répéter jusqu'à "eau bout est vrai"  
    attendre 20 secondes  
  ajouter les pâtes  
  attendre 10 mn  
  égoutter et servir  
Fin procédure
```



Flowchart
ou diagramme de flux

ÉCRIRE UN ALGORITHME ? NIVEAUX DE LANGAGE

- De nombreux langages permettent d'écrire les algorithmes :
 - le langage naturel, le pseudocode, les diagramme de flux
puis les nombreux langages de programmation

le français

```
Si la moyenne atteint ou dépasse  
70, afficher "vous avez réussi"  
sinon, afficher "note insuffisante"
```

ECRIRE UN ALGORITHME ? NIVEAUX DE LANGAGE

- De nombreux langages permettent d'écrire les algorithmes :
 - le langage naturel, le pseudocode, les diagramme de flux
puis les nombreux langages de programmation

le français

```
Si la moyenne atteint ou dépasse  
70, afficher "vous avez réussi"  
sinon, afficher "note insuffisante"
```

le pseudocode :

plus précis et structuré que le langage naturel

```
si moyenne >= 70:  
    mettre texte à "vous avez réussi"  
sinon:  
    mettre texte à "note insuffisante"
```

ECRIRE UN ALGORITHME ? NIVEAUX DE LANGAGE

- De nombreux langages permettent d'écrire les algorithmes :
 - le langage naturel, le pseudocode, les diagramme de flux
puis les nombreux langages de programmation

le français

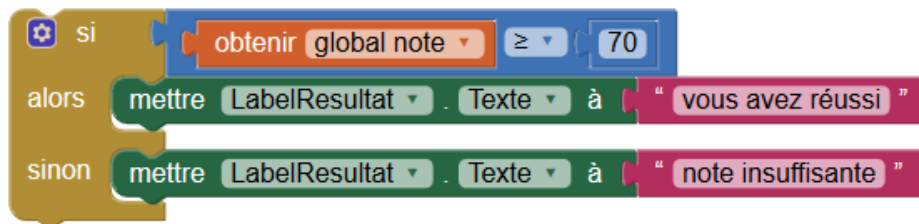
```
Si la moyenne atteint ou dépasse  
70, afficher "vous avez réussi"  
sinon, afficher "note insuffisante"
```

le pseudocode :

plus précis et structuré que le langage naturel

```
si moyenne >= 70:  
    mettre texte à "vous avez réussi"  
sinon:  
    mettre texte à "note insuffisante"
```

Les langages de programmation : scratch, app inventor, javascript, Python



ECRIRE UN ALGORITHME ? NIVEAUX DE LANGAGE

- De nombreux langages permettent d'écrire les algorithmes :
 - le langage naturel, le pseudocode, les diagramme de flux
puis les nombreux langages de programmation

le français

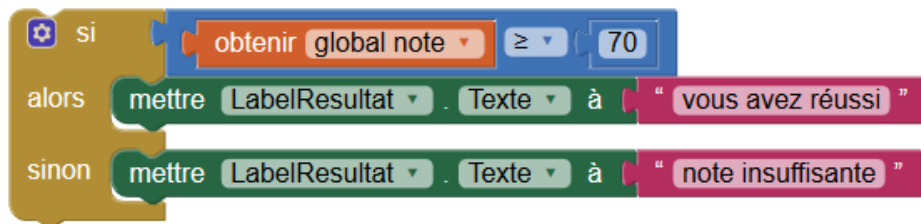
```
Si la moyenne atteint ou dépasse  
70, afficher "vous avez réussi"  
sinon, afficher "note insuffisante"
```

le pseudocode :

plus précis et structuré que le langage naturel

```
si moyenne >= 70:  
    mettre texte à "vous avez réussi"  
sinon:  
    mettre texte à "note insuffisante"
```

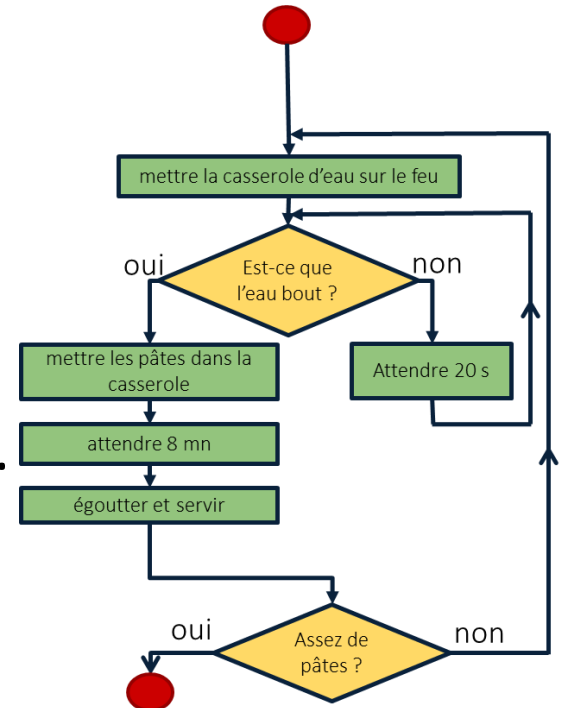
Les langages de programmation : scratch, app inventor, javascript, Python



```
if (moyenne >= 70){  
    evaluation ="vous avez réussi"  
}else{|  
    evaluation ="note insuffisante"  
}  
document.getElementById("demo").innerHTML = evaluation;
```

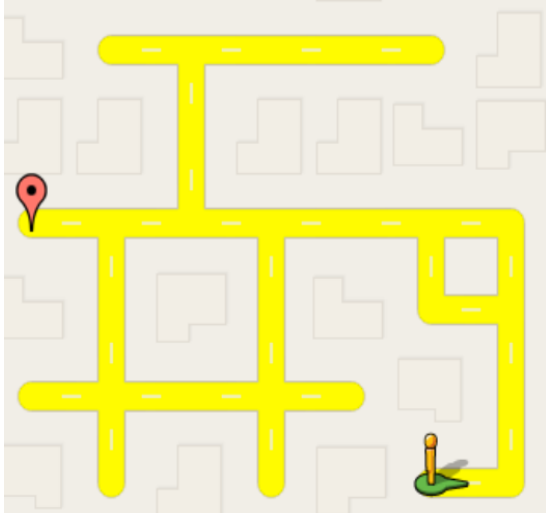
CARACTÉRISTIQUES ET STRUCTURE D'UN ALGORITHME

- Un algorithme est une procédure définie pas-à pas pour résoudre un problème ou obtenir un résultat.
- Un algorithme doit être précis : sans ambiguïté
 - Chaque pas ou étape doit être faisable.
- Tous les algorithmes peuvent être réalisés à partir de séquences, de sélections et de répétitions.
- Les algorithmes peuvent être exprimés par des diagrammes, du pseudocode, ou de nombreux langages de programmation.



3

Exercices écrire l'algorithme de sortie d'un labyrinthe

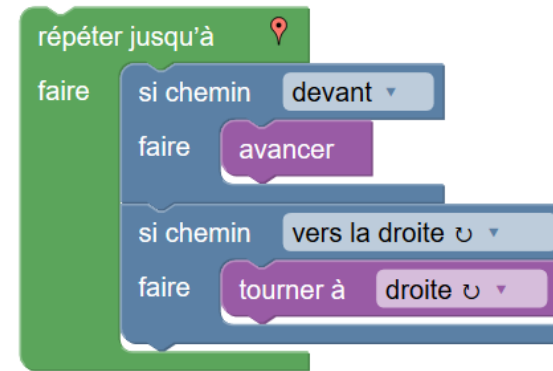
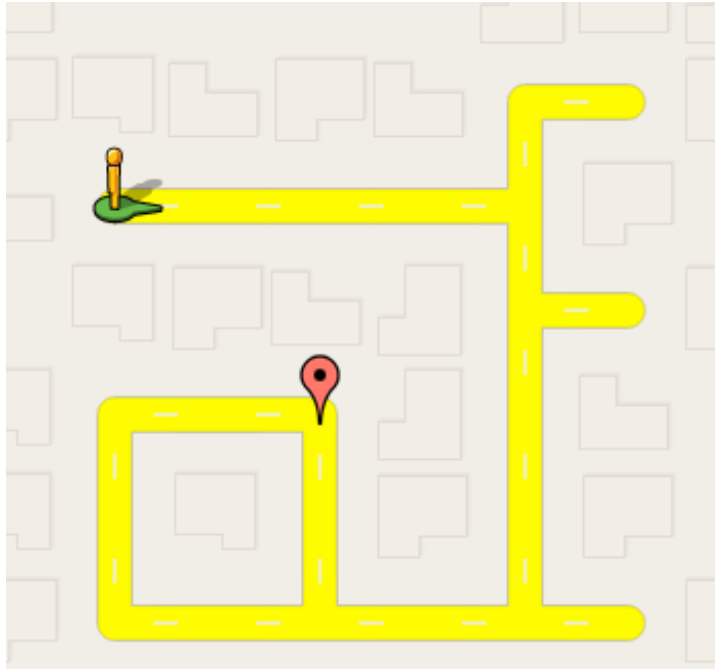


Pseudo-code

Répéter jusqu'à arrivée

```
si chemin à gauche
| tourner à gauche
sinon
| si chemin devant
| |
| | sinon
| | si chemin à droite
| | | tourner à droite
| | sinon
| | | tourner à droite
| | | tourner à droite
| avancer
```

<https://blockly.games/maze?lang=fr>



Félicitations !

Vous avez résolu ce niveau avec 8 lignes de JavaScript :

```
while (notDone()) {  
  if (isPathForward()) {  
    moveForward();  
  }  
  if (isPathRight()) {  
    turnRight();  
  }  
}
```

<https://blockly.games/maze?lang=fr>

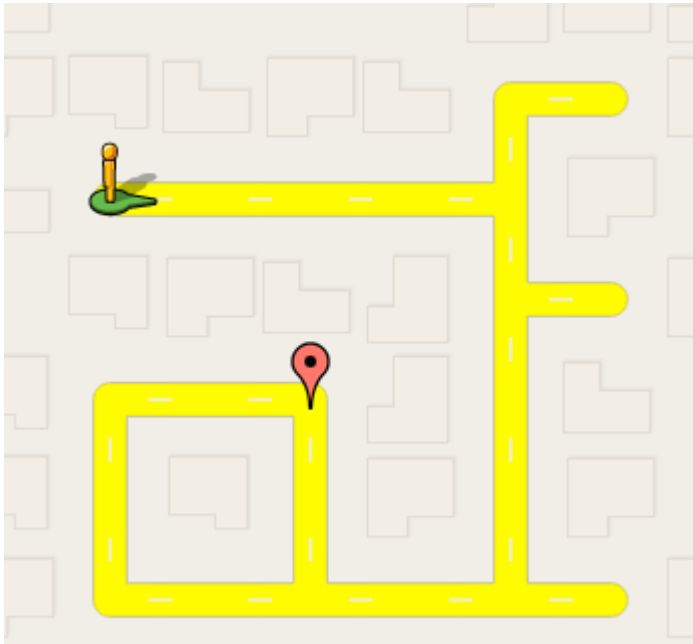
EXERCICE : AVEC LE JEU BLOCKLY MAZE

- Si vous ne connaissez pas ce jeu algorithmique, allez d'abord à l'adresse :
 - <https://blockly.games/maze?lang=fr> et faites les 5 ou 6 premiers niveaux
- Puis allez à l'adresse
 - <https://blockly.games/maze?lang=fr&level=9> (niveau moyen)
 - <https://blockly.games/maze?lang=fr&level=10> (niveau **difficile**)
- Puis dessinez le diagramme des flux, ou écrivez le pseudo-code de l'algorithme qui permet au personnage d'atteindre la sortie en utilisant
 - les instructions du langage : avancer, tourner à gauche, tourner à droite
 - la commande répétition : répéter jusqu'à arrivée
 - la ou les conditions : si chemin devant (ou à gauche ou à droite)
et si chemin devant ... sinon ...



EXERCICE : AVEC LE JEU BLOCKLY MAZE

- Exemple de solution pour le niveau 7



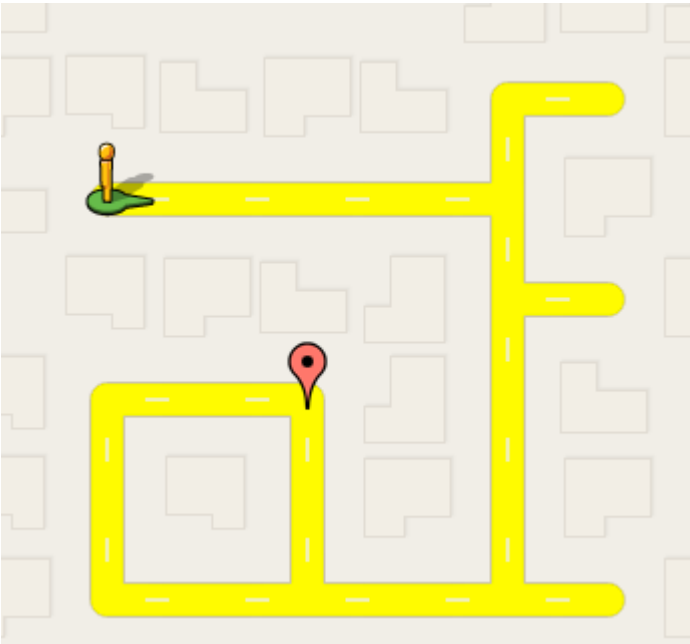
```
avancer
tourner à gauche ↶
tourner à droite ↷
répéter jusqu'à
faire
si chemin devant
faire
si chemin devant
faire
sinon
```

Pseudo-code

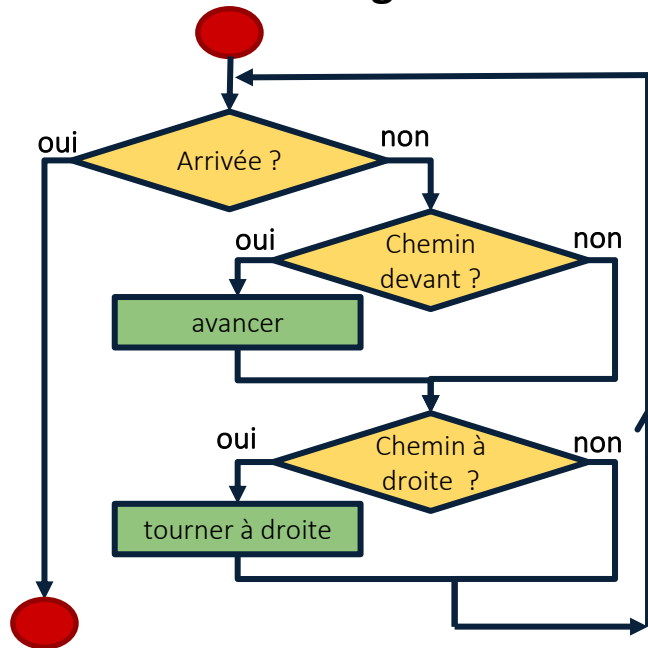
```
Répéter jusqu'à arrivée
| si chemin devant
| | avancer
| si chemin à droite
| | tourner à droite
```

EXERCICE : AVEC LE JEU BLOCKLY MAZE

- Exemple de solution pour le niveau 7



Diagramme



Pseudo-code

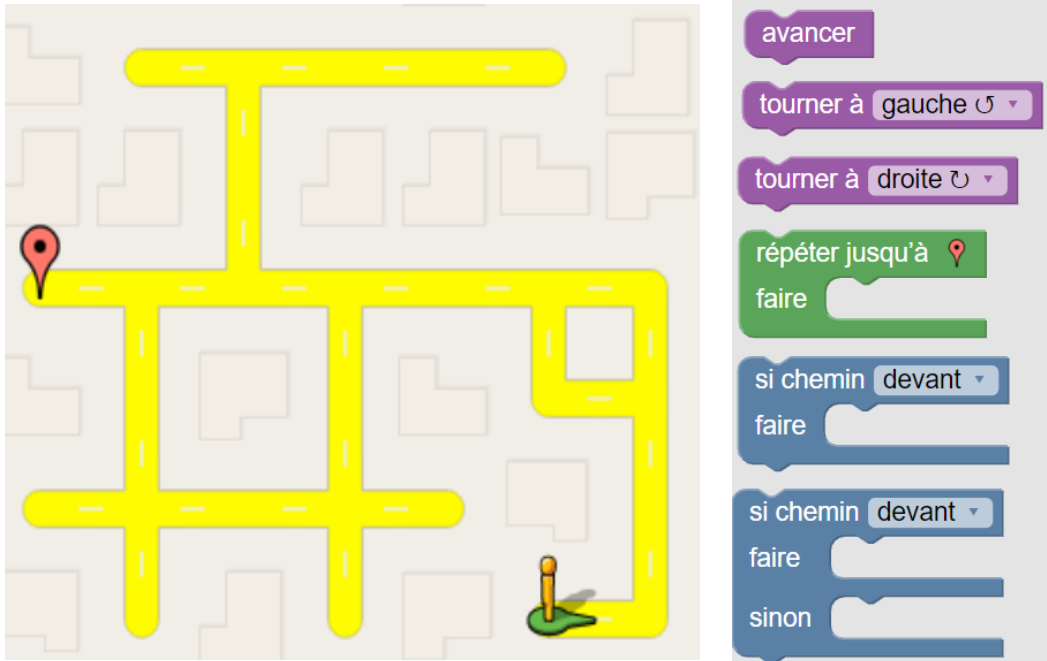
Répéter jusqu'à arrivée
| si chemin devant
| | avancer
| si chemin à droite
| | tourner à droite

EXERCICE : ALGORITHME DE SORTIE DU LABYRINTHE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

Problème à traiter et Instructions disponibles



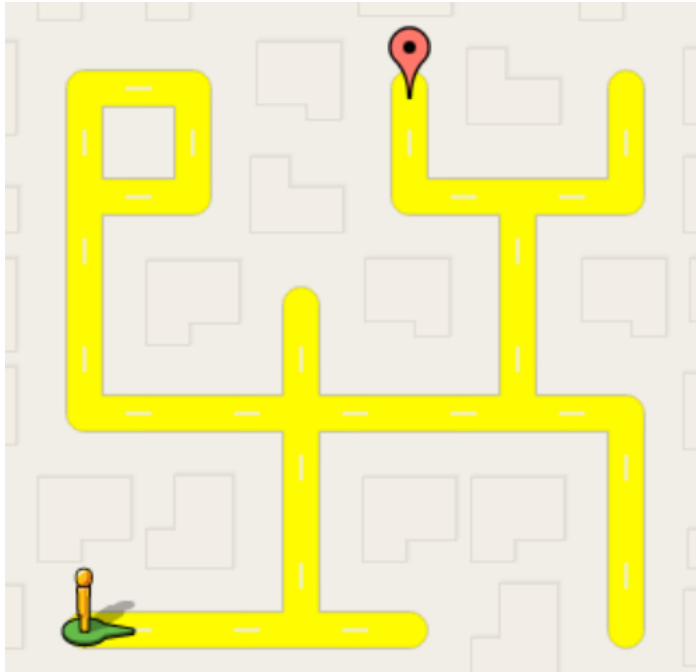
- Écrivez en français comment vous allez faire, quelle est votre stratégie ?
- ensuite écrivez le pseudo-code
- en option, le diagramme des flux
- Vous avez besoin d'un crayon et d'un papier. N'utilisez PAS l'ordinateur
- Vous devez :
 - commencer par "répéter jusqu'à arrivée"
 - utiliser les termes "avancer", "tourner"
 - Et les conditions si chemin devant, à gauche, à droite ou la condition si ... sinon ...

EXERCICE : ALGORITHME DE SORTIE DU LABYRINTHE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Problème à traiter et Instructions disponibles



Les instructions sont les mêmes que pour le niveau 9, mais c'est plus difficile.

N'utilisez qu'un crayon et un papier

La phrase en français qui décrit l'algorithme dont vous devez écrire le pseudo-code est :

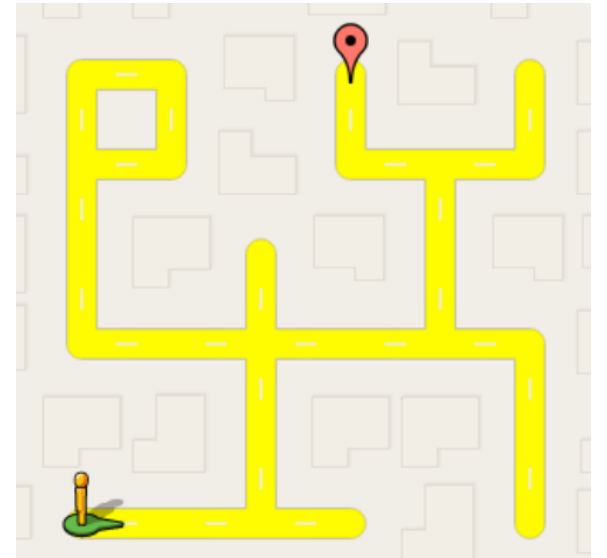
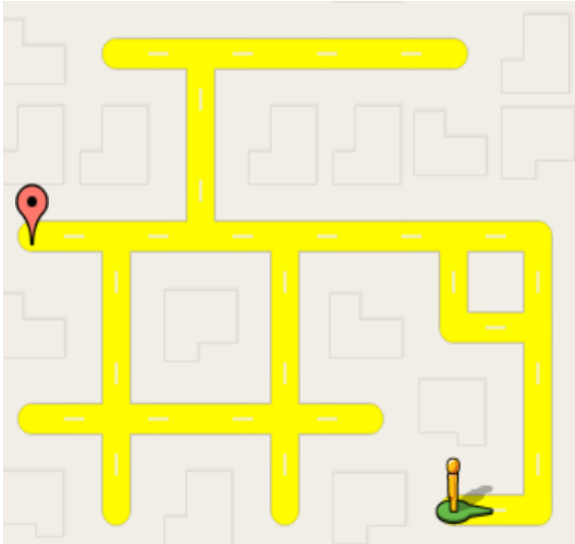
- Longer le mur de gauche jusqu'à la sortie

Ca correspond simplement à dire que si vous longez le mur, vous finirez bien par arriver à la porte de sortie!

Ecrivez le pseudocode qui longe le mur.

Exercices pseudo-code

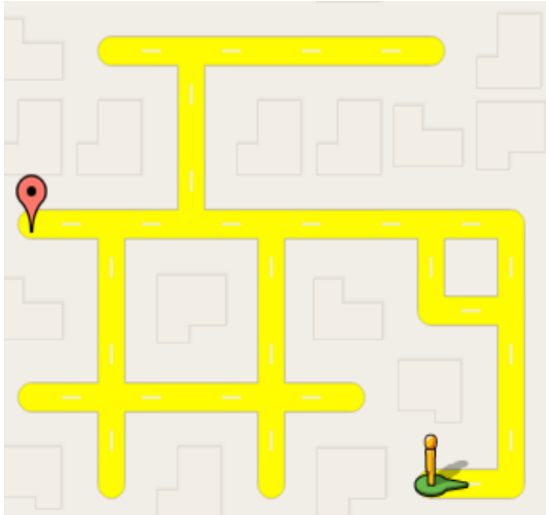
à vous de jouer



Vidéo suivante : <https://onvaessayer.org?video=algoSolutions>

3b

Solution des exercices écrire l'algorithme de sortie d'un labyrinthe



Pseudo-code

Répéter jusqu'à arrivée

```
si chemin à gauche
| tourner à gauche
sinon
| si chemin devant
| |
| | sinon
| | si chemin à droite
| | | tourner à droite
| | sinon
| | | tourner à droite
| | | tourner à droite
avancer
```

Pierre Huguet

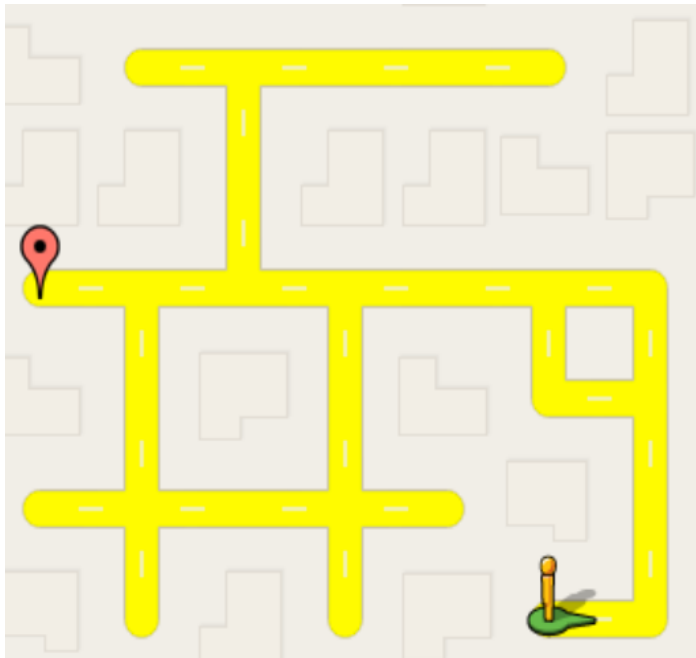
EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

Pseudo-code

Problème à traiter et Instructions disponibles



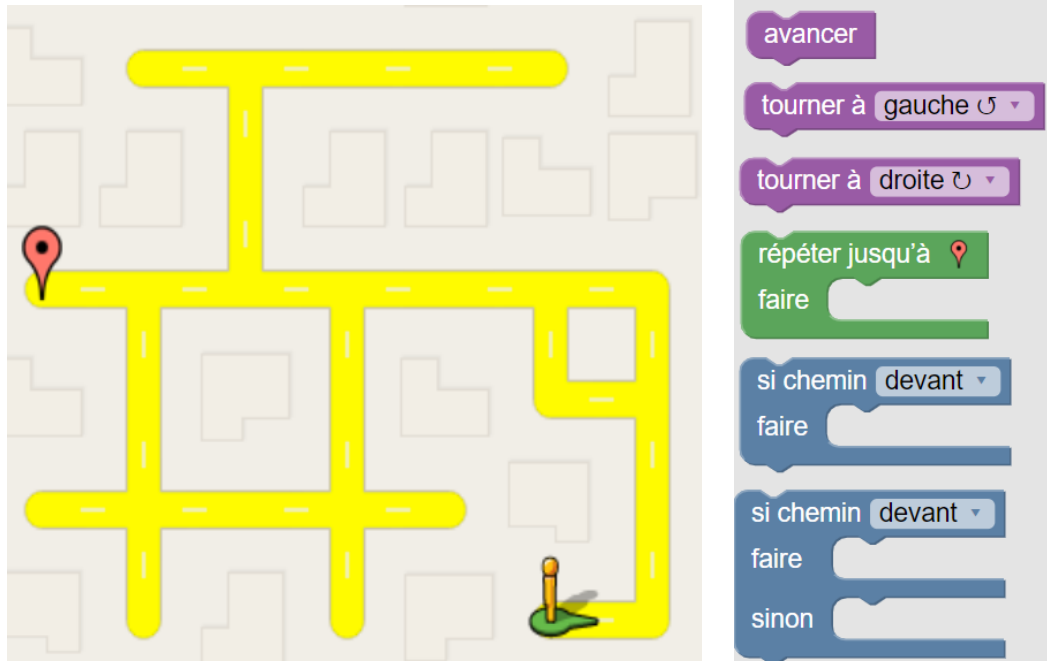
```
graph TD; A[avancer] --> B[tourner à gauche ↶]; B --> C[tourner à droite ↷]; C --> D[répéter jusqu'à 📍]; D --> E[faire]; E --> F[si chemin devant ▼]; F --> G[faire]; G --> H[si chemin devant ▼]; H --> I[faire]; I --> J[sinon];
```

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

Problème à traiter et Instructions disponibles



Pseudo-code niveau 7

Répéter jusqu'à arrivée

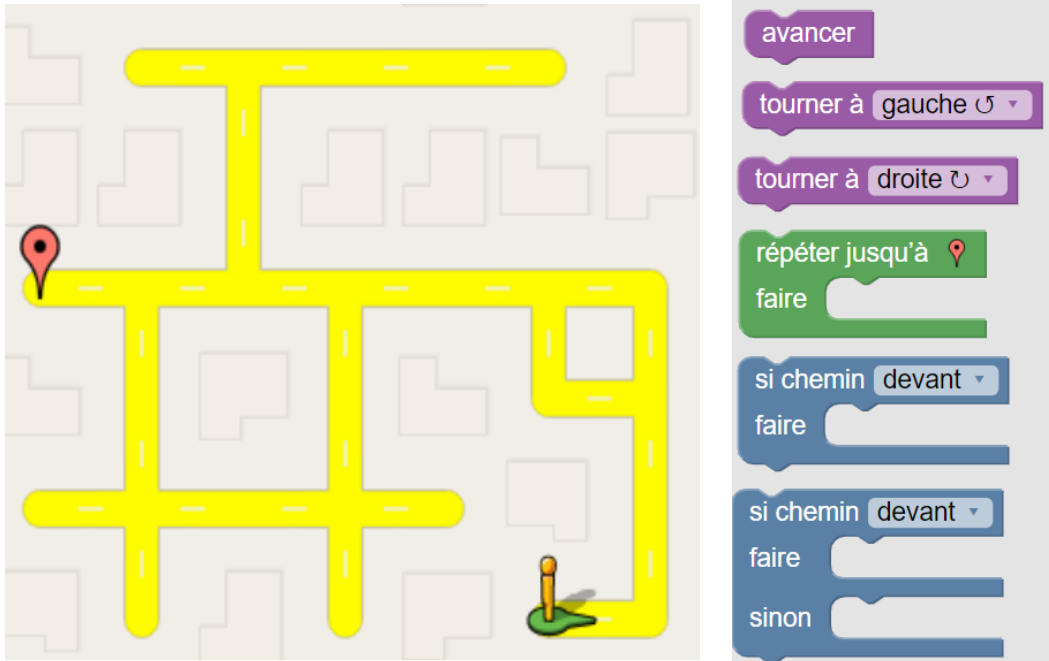
```
si chemin devant  
| avancer  
si chemin à droite  
| tourner à droite
```

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

Problème à traiter et Instructions disponibles



Pseudo-code niveau 9

Répéter jusqu'à arrivée

```
si chemin devant
  avancer
si chemin à gauche
  tourner à gauche
```

Pseudo-code niveau 7

Répéter jusqu'à arrivée

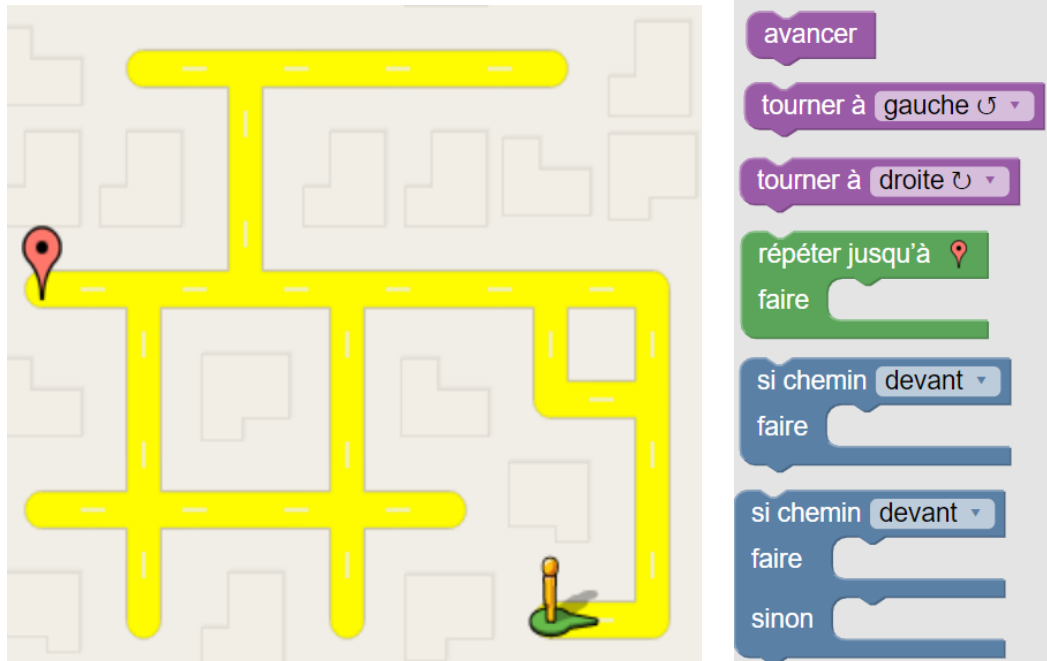
```
si chemin devant
  avancer
si chemin à droite
  tourner à droite
```

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

Problème à traiter et Instructions disponibles



Pseudo-code niveau 9

Répéter jusqu'à arrivée

si chemin devant

avancer

si chemin à gauche

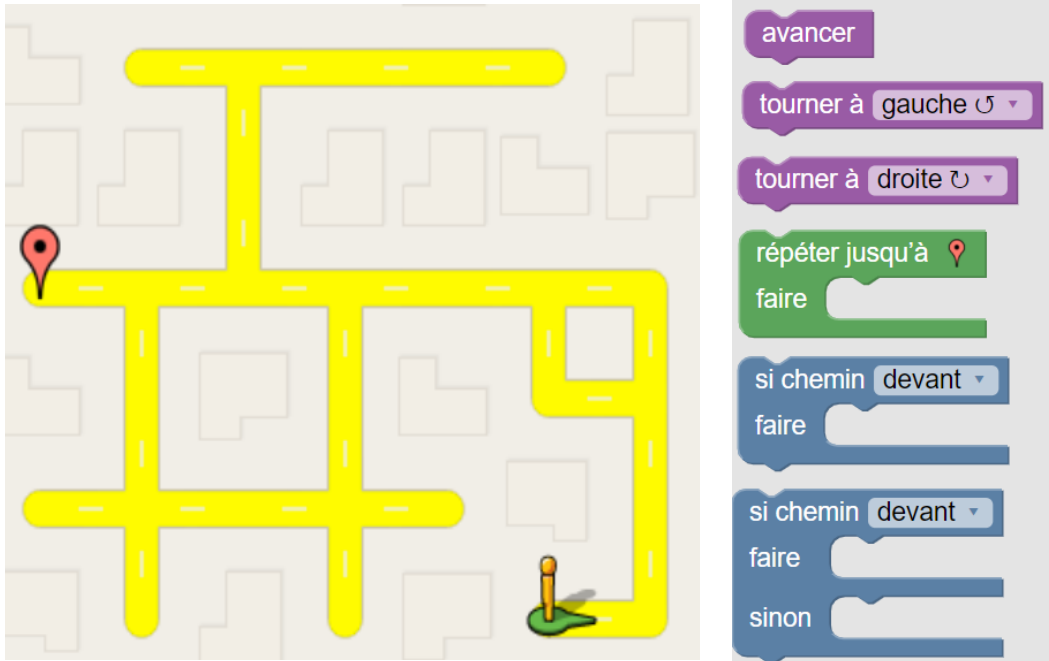
tourner à gauche

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

Problème à traiter et Instructions disponibles



Pseudo-code niveau 9

```
Répéter jusqu'à arrivée  
| si chemin devant  
| | avancer  
| si chemin à gauche  
| | tourner à gauche
```

Pseudo-code niveau 9

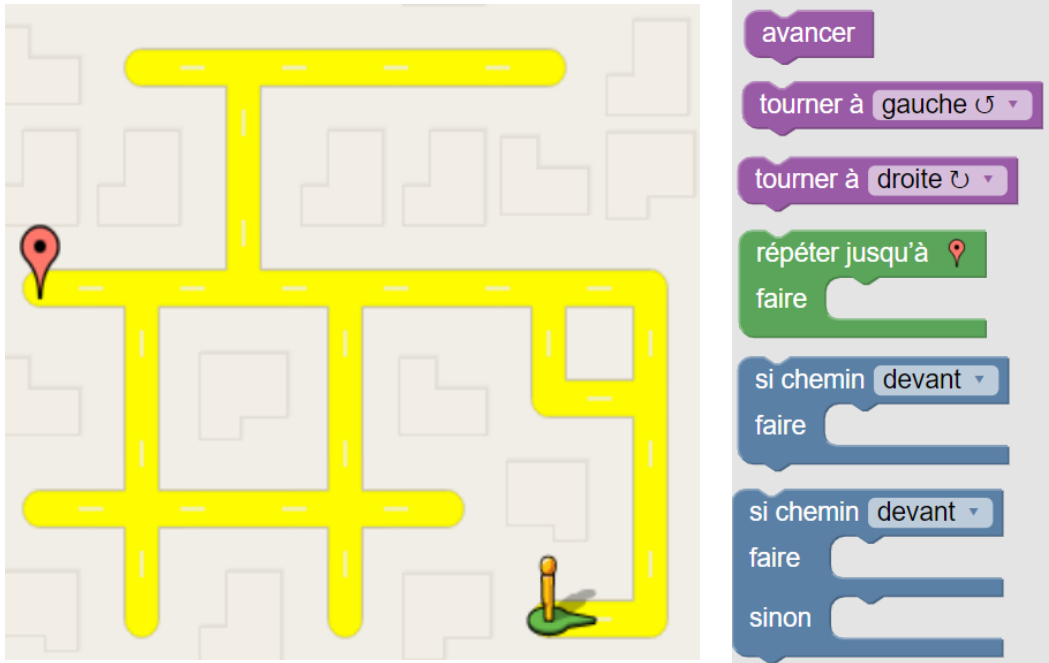
```
Répéter jusqu'à arrivée  
| si chemin devant  
| | avancer  
| si
```

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

Problème à traiter et Instructions disponibles



Pseudo-code niveau 9

~~Répéter jusqu'à arrivée
| si chemin devant
| | avancer
| si chemin à gauche
| | tourner à gauche~~

Pseudo-code niveau 9

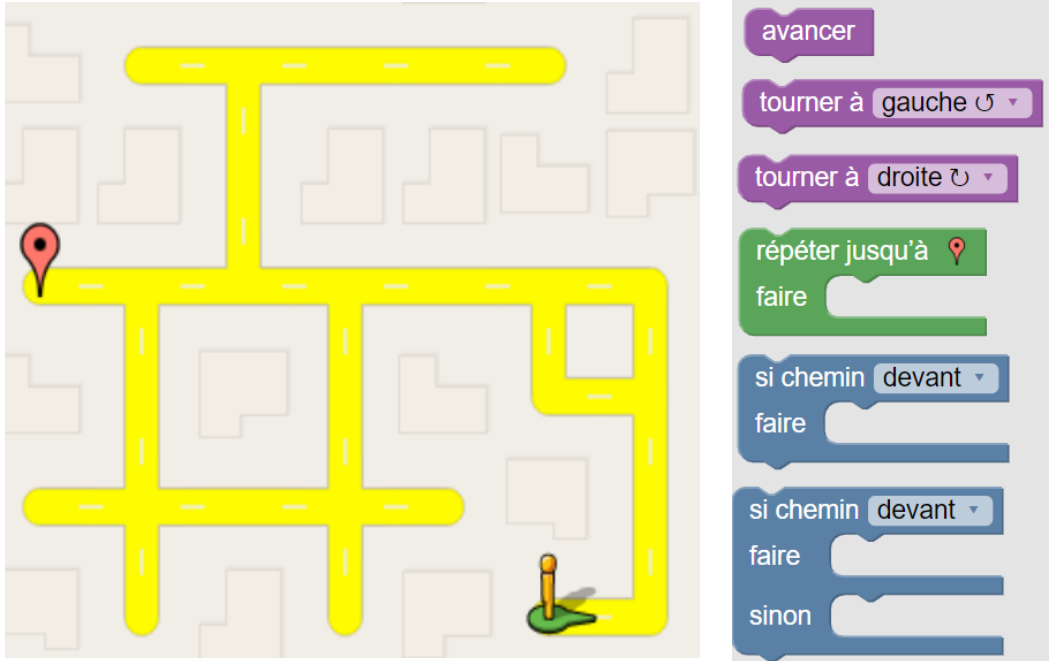
Répéter jusqu'à arrivée
| si chemin devant
| | avancer
| sinon
| | tourner à gauche

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

Problème à traiter et Instructions disponibles

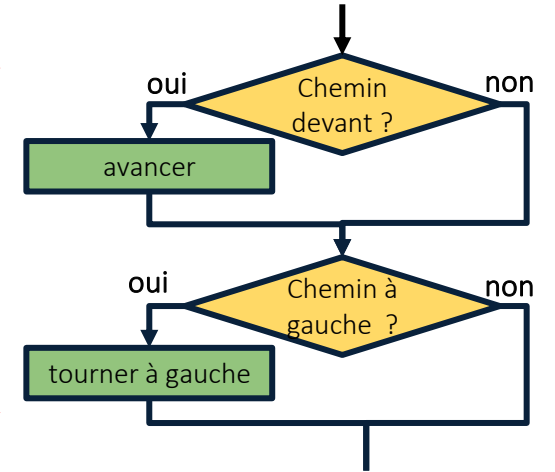


Pseudo-code niveau 9

~~Répéter jusqu'à arrivée
si chemin devant
| avancer
si chemin à gauche
| tourner à gauche~~

Pseudo-code niveau 9

Répéter jusqu'à arrivée
| si chemin devant
| | avancer
sinon
| | tourner à gauche

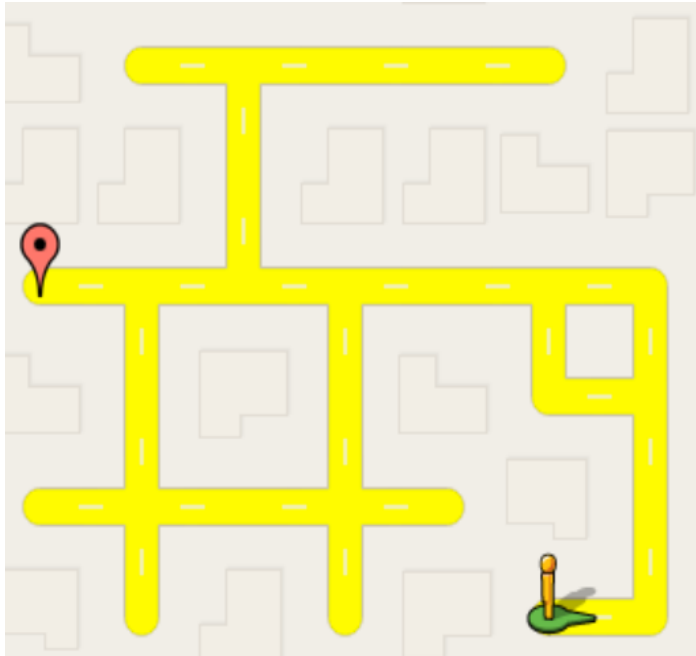


EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

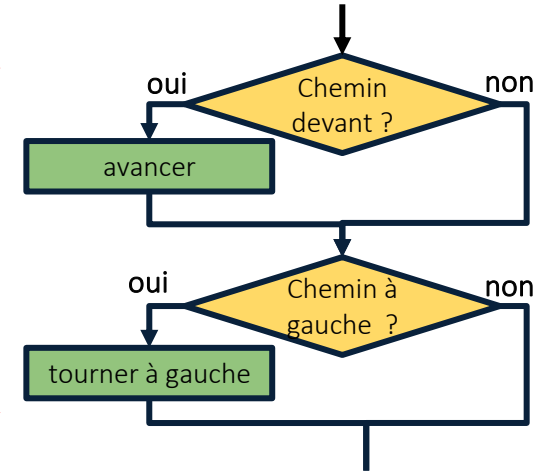
Problème à traiter et Instructions disponibles



```
graph TD
    A[avancer]
    B[tourner à gauche]
    C[tourner à droite]
    D[répéter jusqu'à]
    E[si chemin devant]
    F[sinon]
```

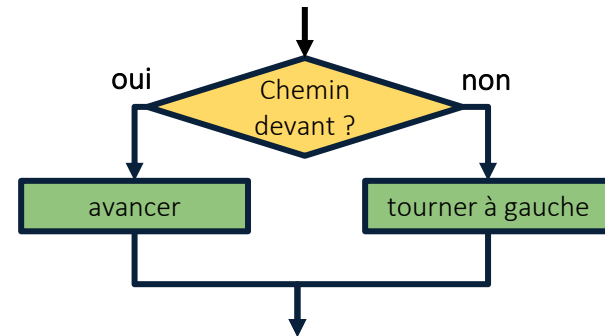
~~Pseudo-code niveau 9~~

~~Répéter jusqu'à arrivée
si chemin devant
| avancer
si chemin à gauche
| tourner à gauche~~



Pseudo-code niveau 9

Répéter jusqu'à arrivée
si chemin devant
| avancer
sinon
| tourner à gauche

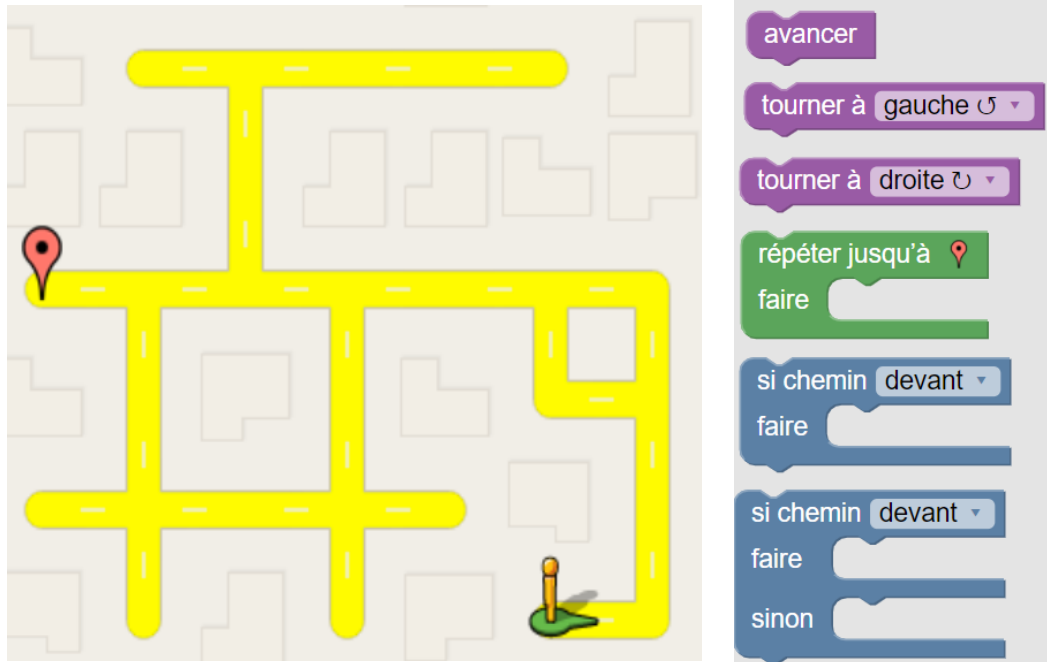


EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N9

Blockly maze, niveau 9 :

<https://blockly.games/maze?lang=fr&level=9>

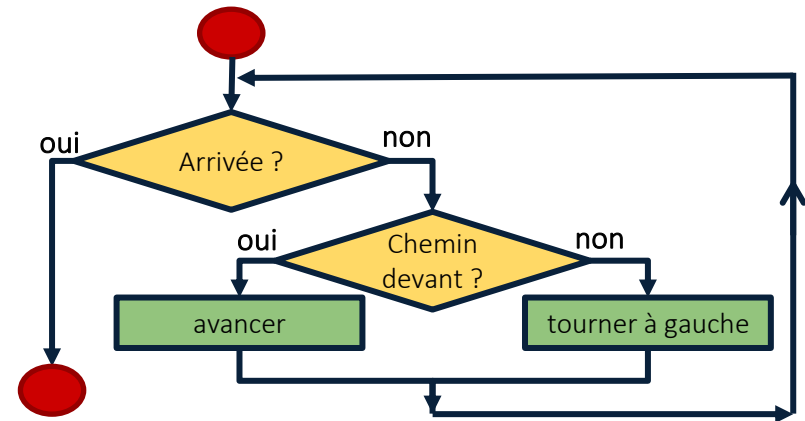
Problème à traiter et Instructions disponibles



Pseudo-code

Répéter jusqu'à arrivée
| si chemin devant
| | avancer
| sinon
| | tourner à gauche

Diagramme



EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

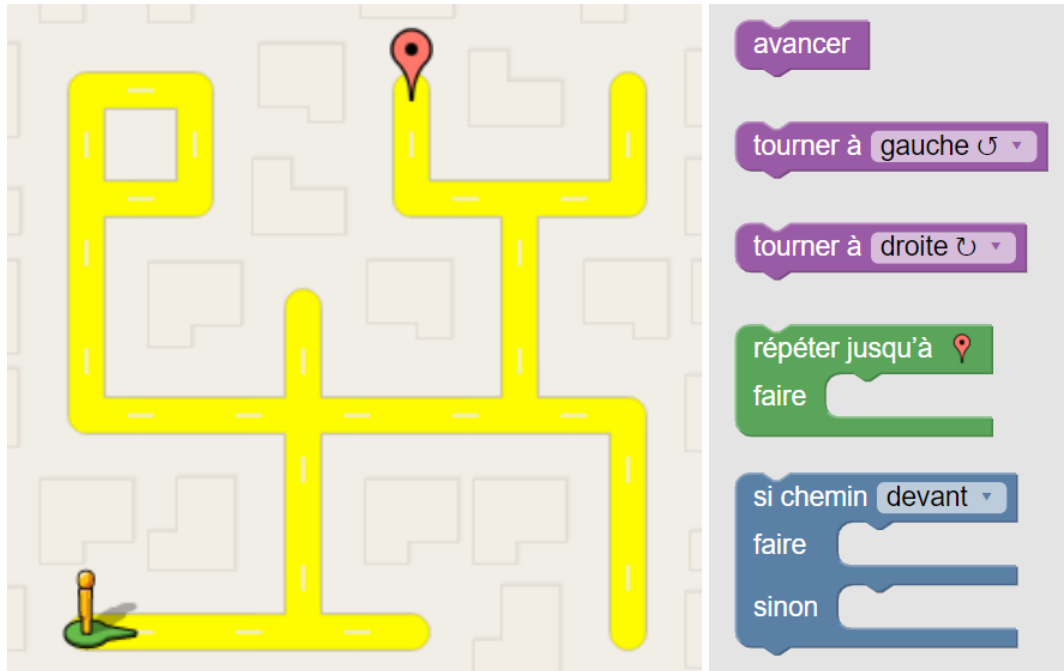
Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Phrase en français qui décrit l'algorithme :

Longer le mur de gauche jusqu'à la sortie

Problème à traiter et Instructions disponibles



The image shows a screenshot of a Blockly maze game interface. On the left, a maze is displayed with a yellow path leading from a green starting point (a small figure) to a red pin indicating the exit. The maze consists of several interconnected paths and dead ends. On the right, a list of instructions is available for use in the Blockly editor:

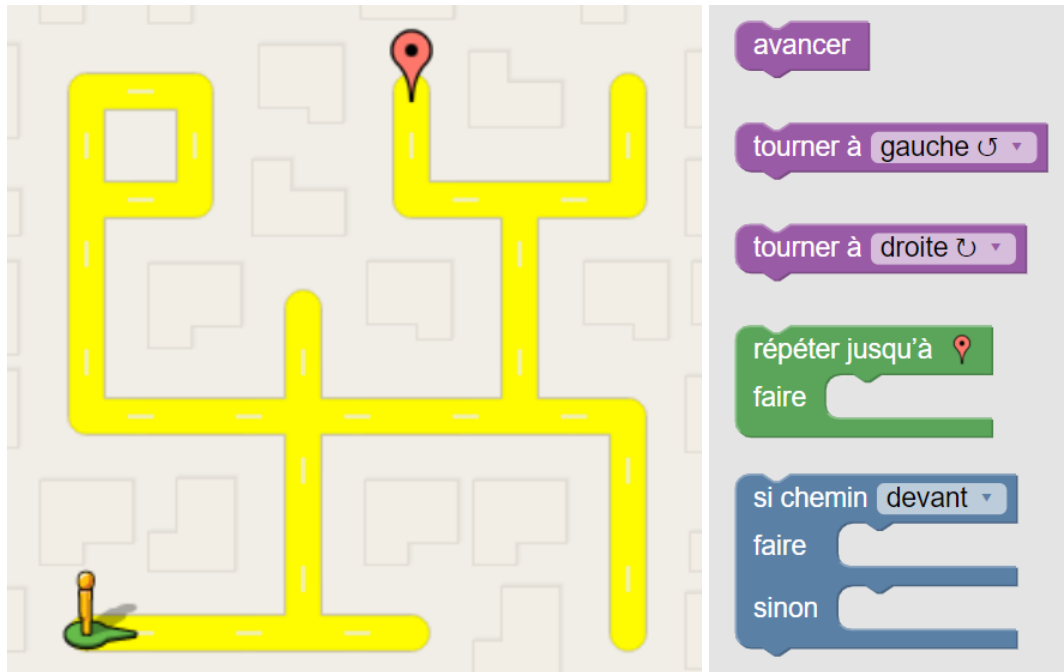
- avancer
- tourner à gauche ↶
- tourner à droite ↷
- répéter jusqu'à [pin icon]
- faire
- si chemin devant
- faire
- sinon

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Problème à traiter et Instructions disponibles



Longer le mur de gauche jusqu'à la sortie

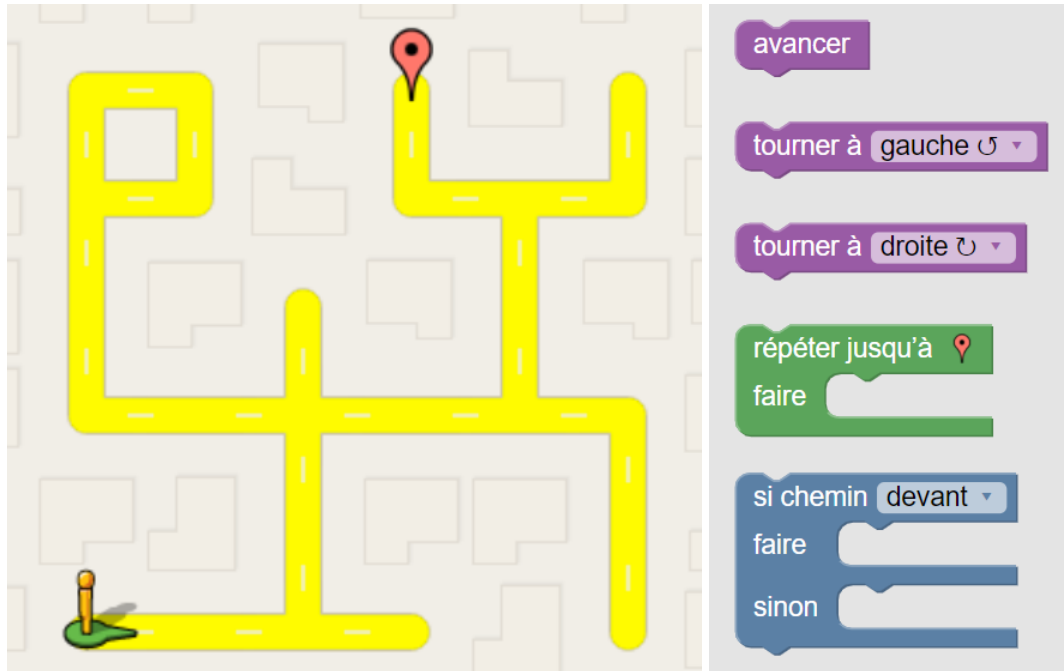
Répéter jusqu'à arrivée

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Problème à traiter et Instructions disponibles



Longer le mur de gauche jusqu'à la sortie

Répéter jusqu'à arrivée

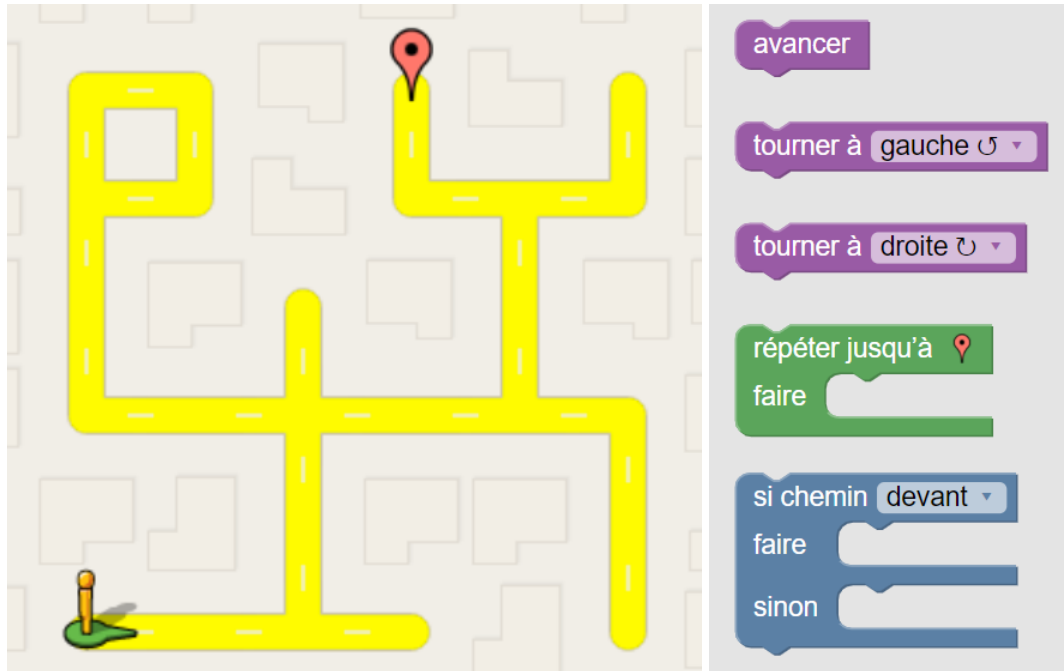
si chemin à gauche
tourner à gauche
avancer

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Problème à traiter et Instructions disponibles



Longer le mur de gauche jusqu'à la sortie

Répéter jusqu'à arrivée

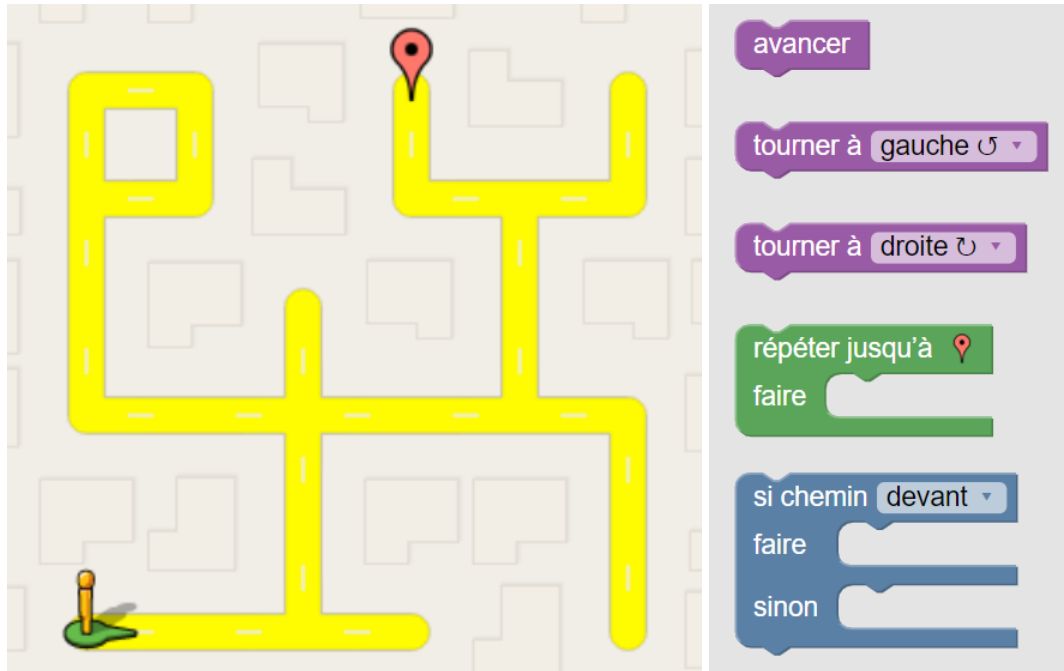
```
si chemin à gauche  
| tourner à gauche  
| avancer  
sinon
```

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Problème à traiter et Instructions disponibles



Longer le mur de gauche jusqu'à la sortie

Répéter jusqu'à arrivée

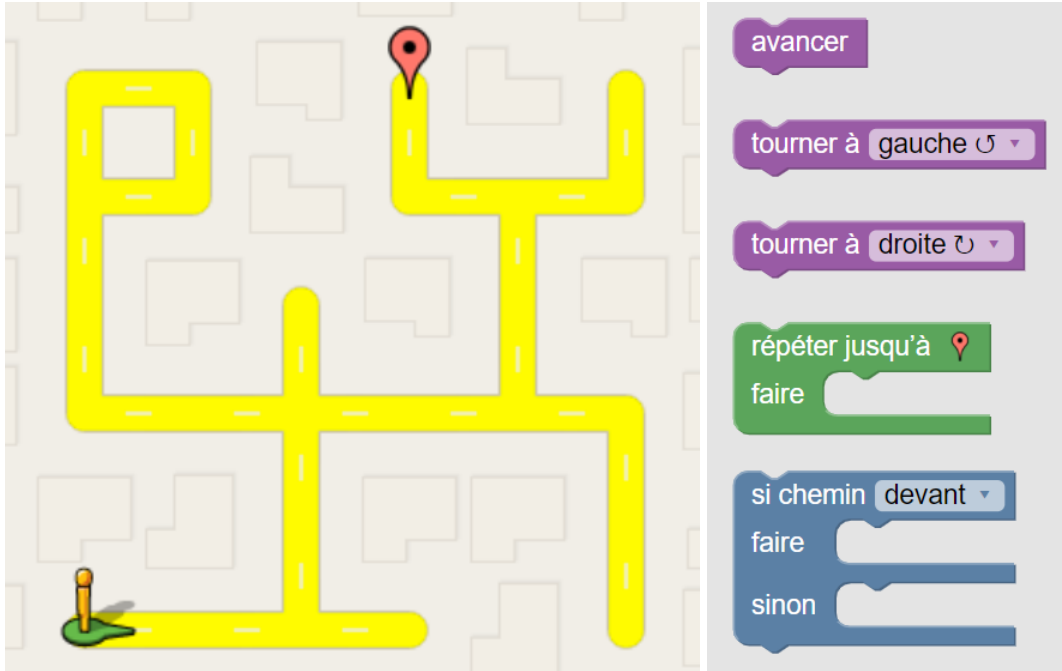
```
si chemin à gauche
| tourner à gauche
| avancer
sinon
| si chemin devant
| | avancer
sinon
```

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Problème à traiter et Instructions disponibles



Longer le mur de gauche jusqu'à la sortie

Répéter jusqu'à arrivée

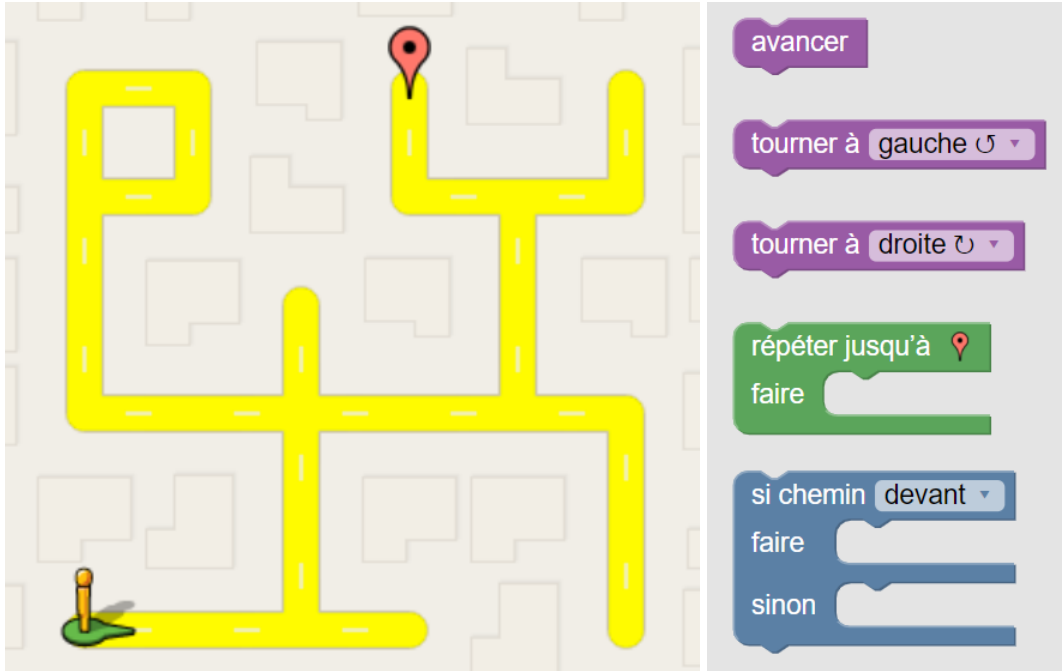
```
si chemin à gauche
| tourner à gauche
| avancer
sinon
| si chemin devant
| | avancer
sinon
| si chemin à droite
| | tourner à droite
| | avancer
sinon
```

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Problème à traiter et Instructions disponibles



Longer le mur de gauche jusqu'à la sortie

Répéter jusqu'à arrivée

si chemin à gauche
| tourner à gauche
| avancer

sinon

| si chemin devant
| | avancer

sinon

| | si chemin à droite
| | | tourner à droite
| | | avancer

sinon

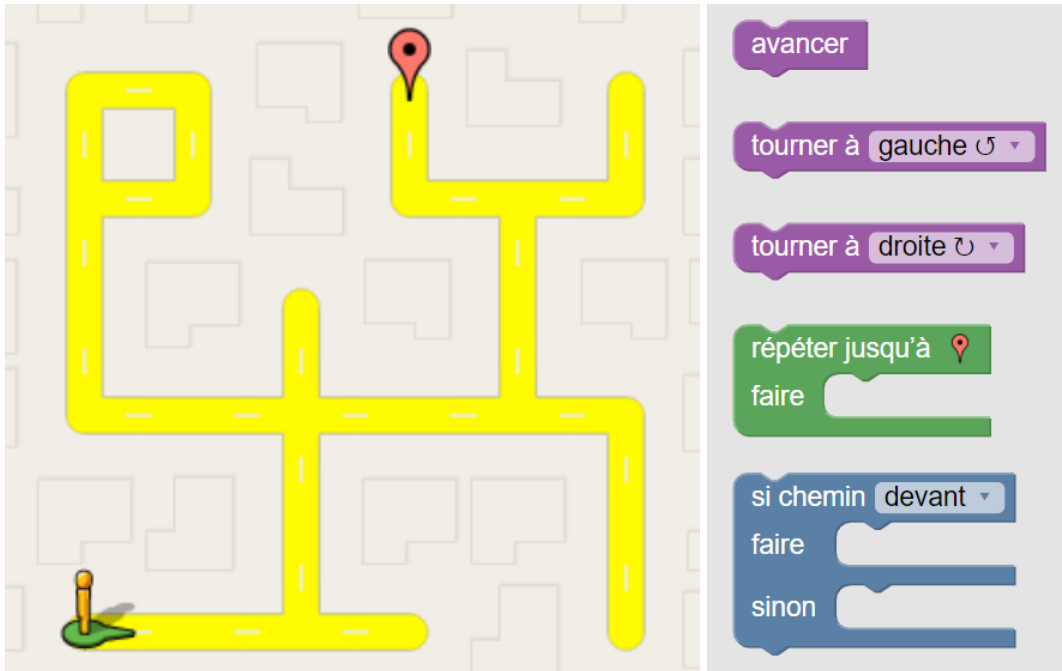
| | | tourner à droite
| | | tourner à droite
| | | avancer

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

Problème à traiter et Instructions disponibles



Longer le mur de gauche jusqu'à la sortie

Répéter jusqu'à arrivée

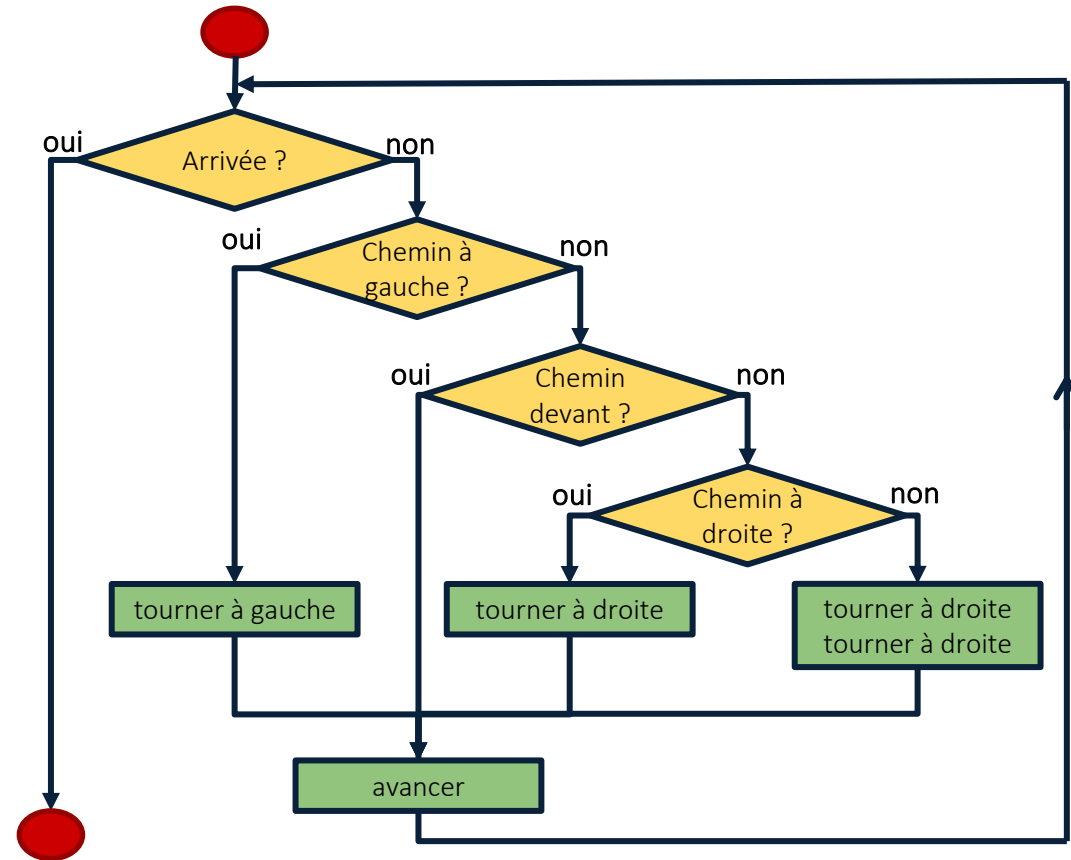
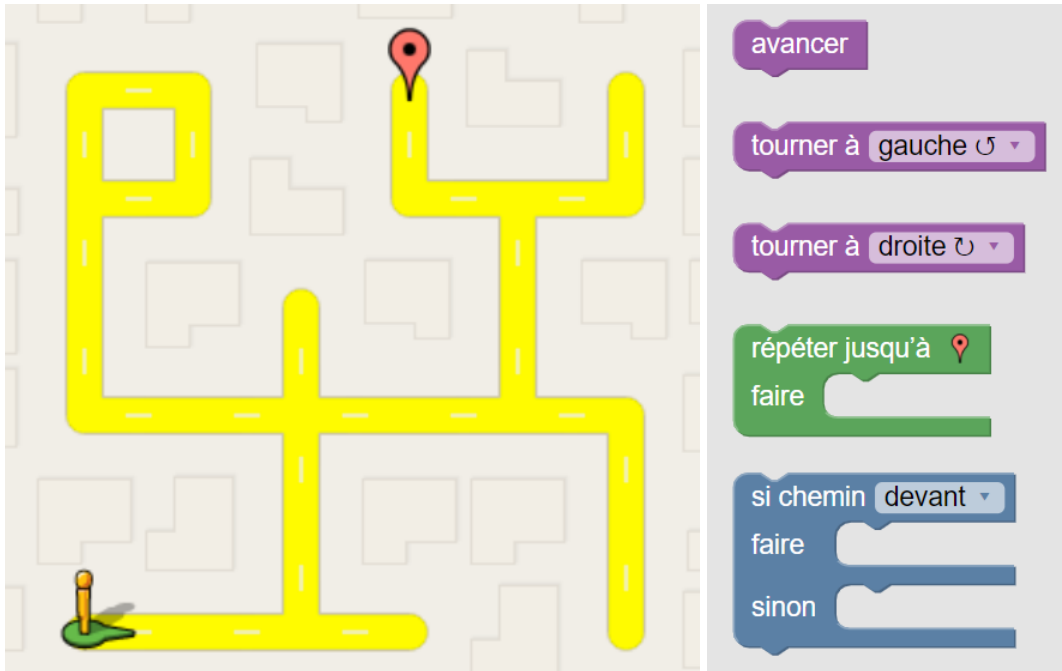
```
si chemin à gauche
|
|  tourner à gauche
|
sinon
|
|  si chemin devant
|  |
|  |  sinon
|  |  |
|  |  |  si chemin à droite
|  |  |  |
|  |  |  |  tourner à droite
|  |  |  |
|  |  |  |  sinon
|  |  |  |  |
|  |  |  |  |  tourner à droite
|  |  |  |  |
|  |  |  |  |  tourner à droite
|  |  |  |  |
|  |  |  |  |  avancer
```

EXERCICE : SOLUTION DE L'ALGORITHME DE SORTIE N10

Blockly maze, niveau 10 :

<https://blockly.games/maze?lang=fr&level=10>

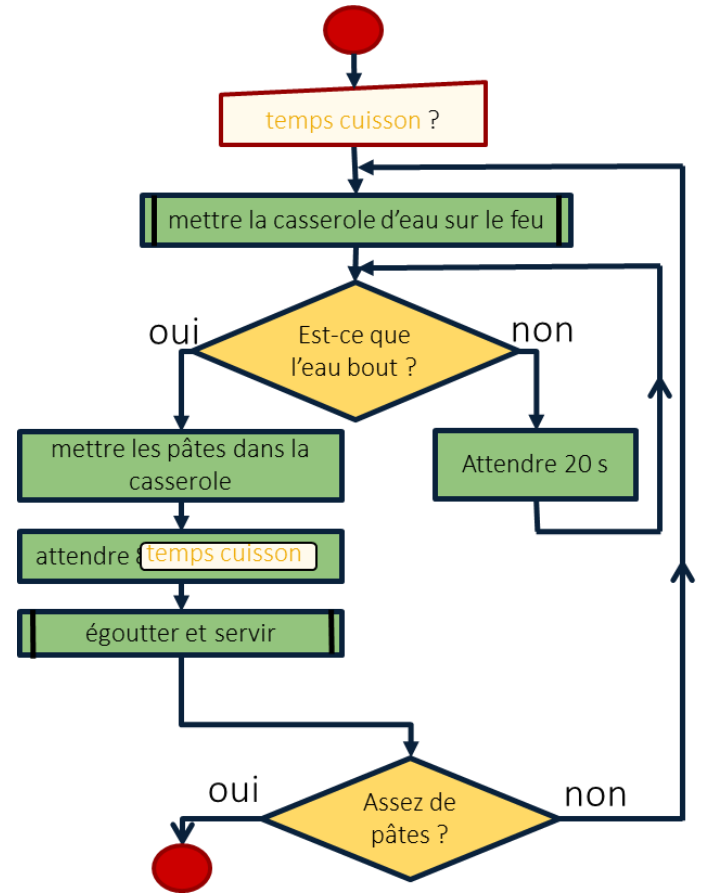
Problème à traiter et Instructions disponibles



4

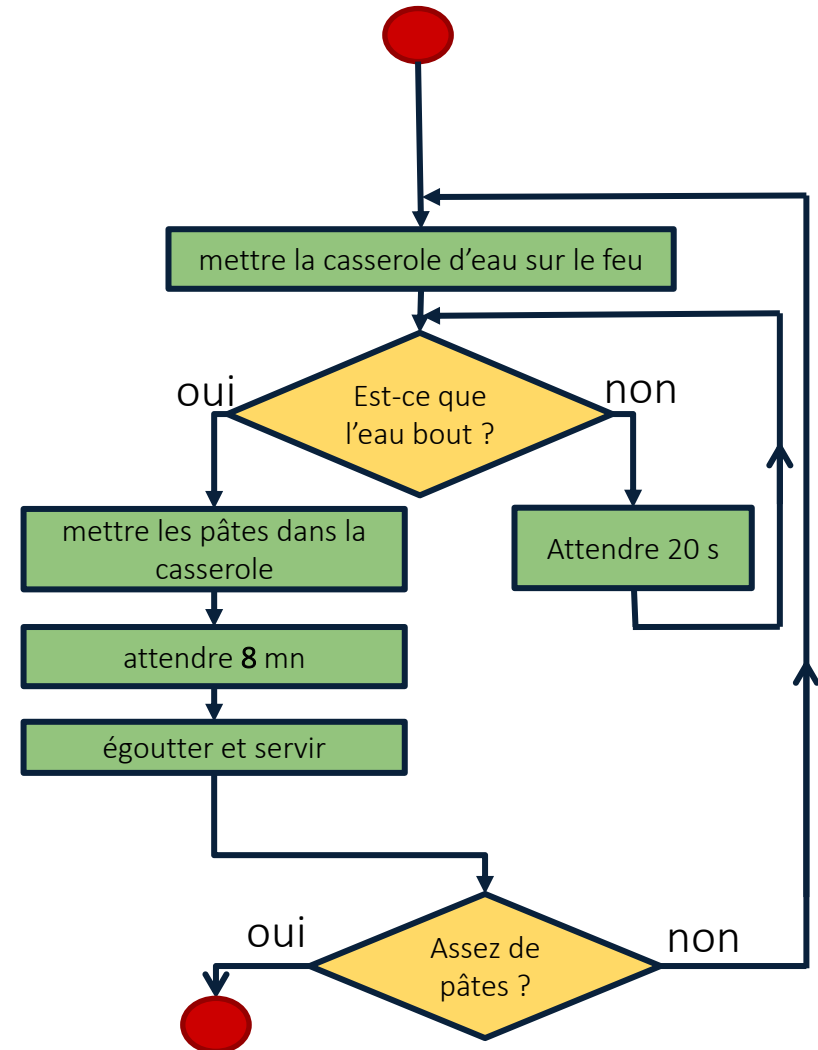
variables et procédures dans un algorithme

Pierre Huguet



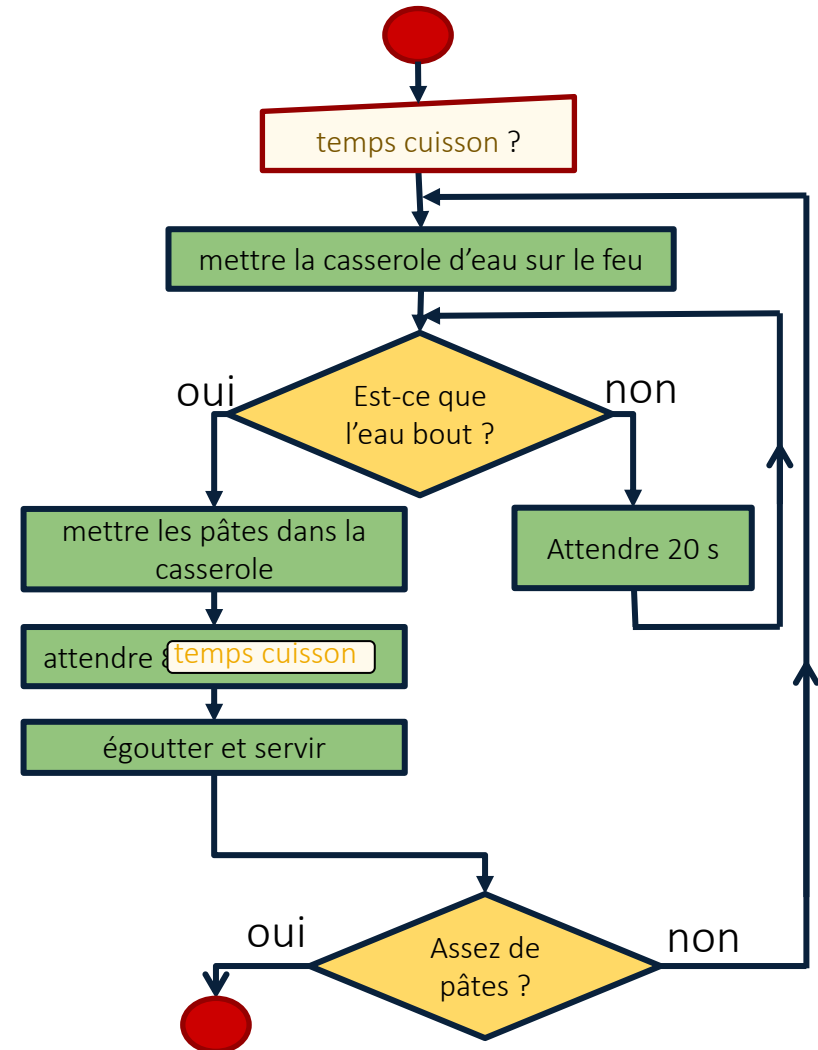
VARIABLES

- Est-ce que l'algorithme des pâtes peut s'adapter au goût des personnes en faisant **varier** le **temps de cuisson** ?



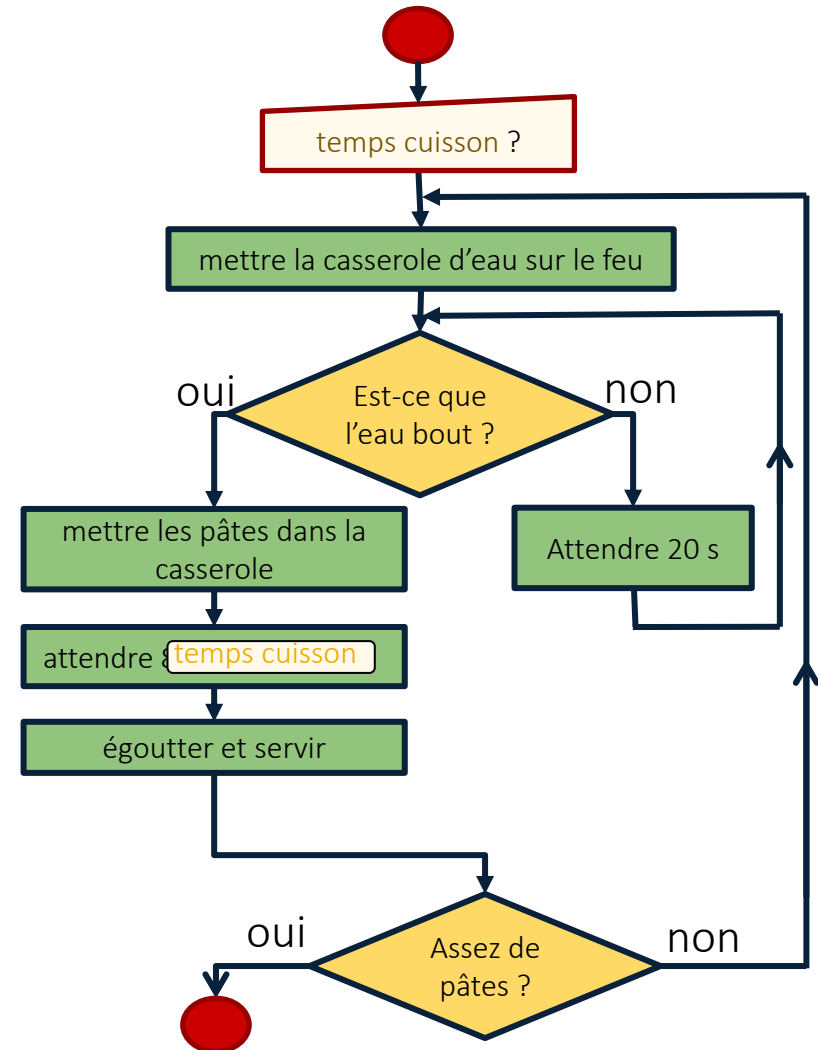
INTRODUCTION AUX VARIABLES

- Est-ce que l'algorithme des pâtes peut s'adapter au goût des personnes en faisant **varier** le **temps de cuisson** ?



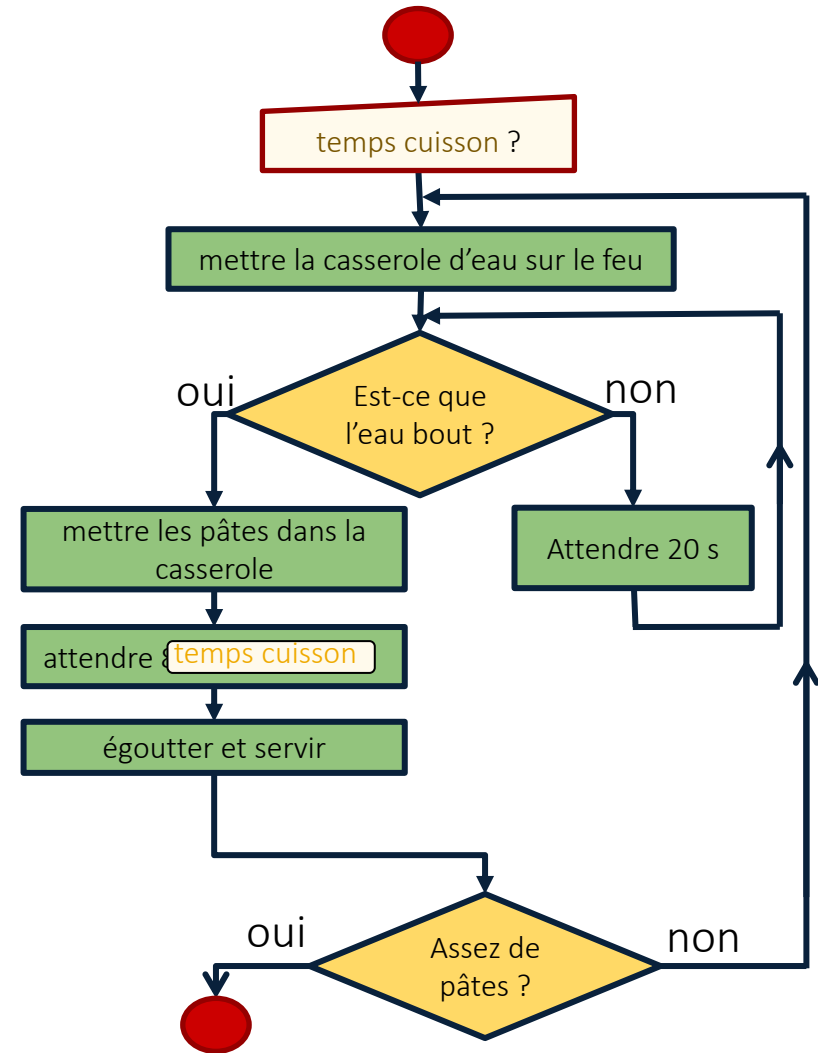
INTRODUCTION AUX VARIABLES

- Est-ce que l'algorithme des pâtes peut s'adapter au goût des personnes en faisant **varier** le **temps de cuisson** ?
- Au début de l'algorithme
 - on crée une variable `temps cuisson`



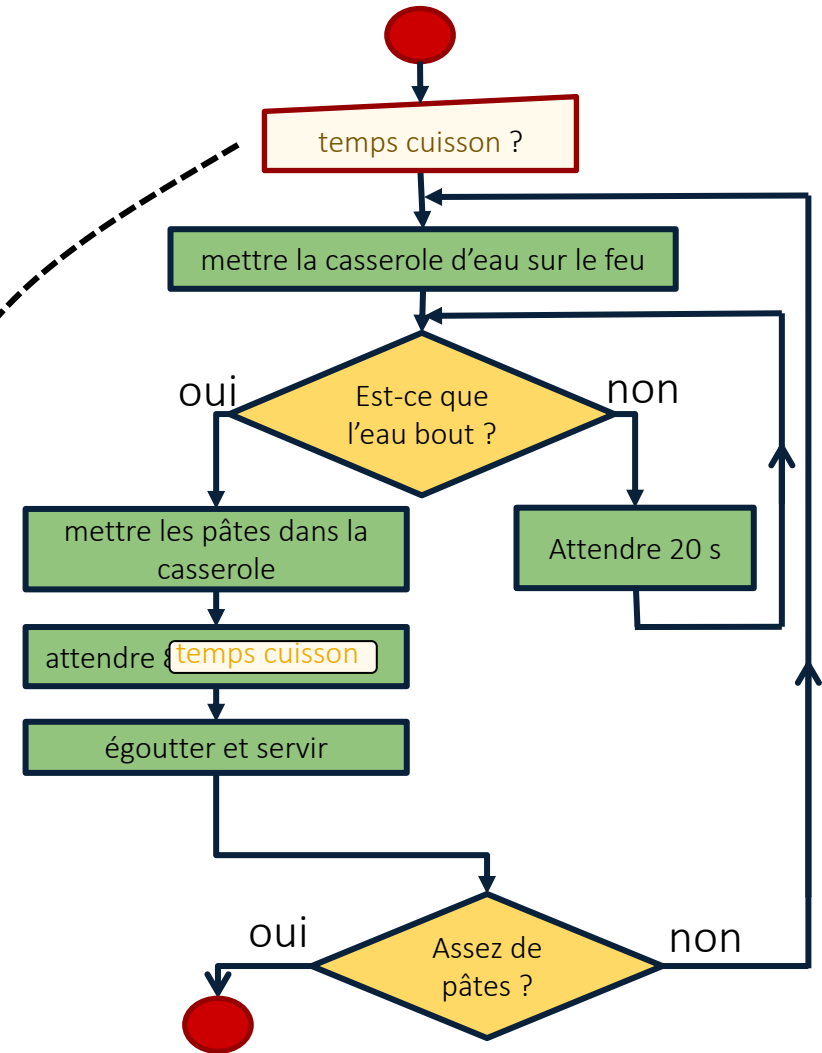
INTRODUCTION AUX VARIABLES

- Est-ce que l'algorithme des pâtes peut s'adapter au goût des personnes en faisant **varier** le **temps de cuisson** ?
- Au début de l'algorithme
 - on crée une variable `temps cuisson`
 - dont on demande la valeur pour l'enregistrer ou la stocker



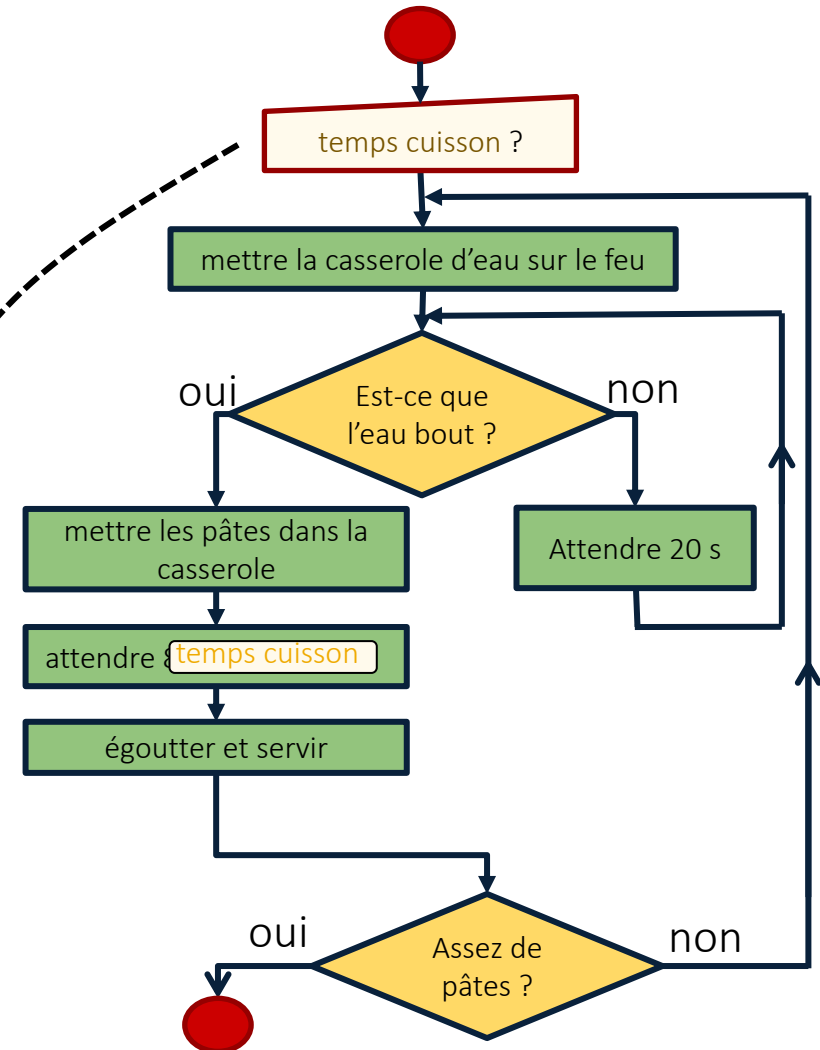
INTRODUCTION AUX VARIABLES

- Est-ce que l'algorithme des pâtes peut s'adapter au goût des personnes en faisant **varier** le **temps de cuisson** ?
- Au début de l'algorithme
 - on crée une variable `temps cuisson`
 - dont on demande la valeur pour l'enregistrer ou la stocker



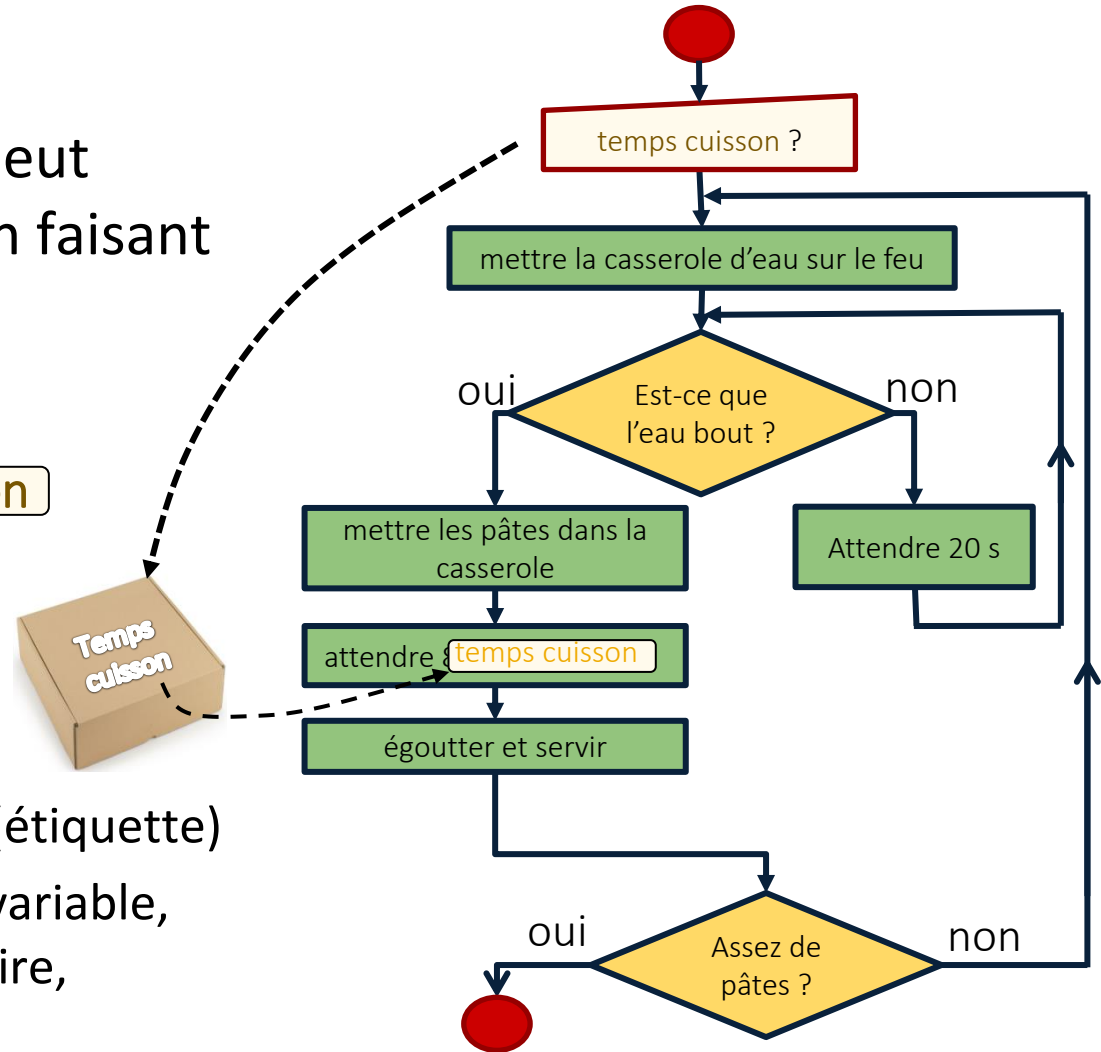
INTRODUCTION AUX VARIABLES

- Est-ce que l'algorithme des pâtes peut s'adapter au goût des personnes en faisant **varier** le **temps de cuisson** ?
- Au début de l'algorithme
 - on crée une variable `temps cuisson`
 - dont on demande la valeur pour l'enregistrer ou la stocker



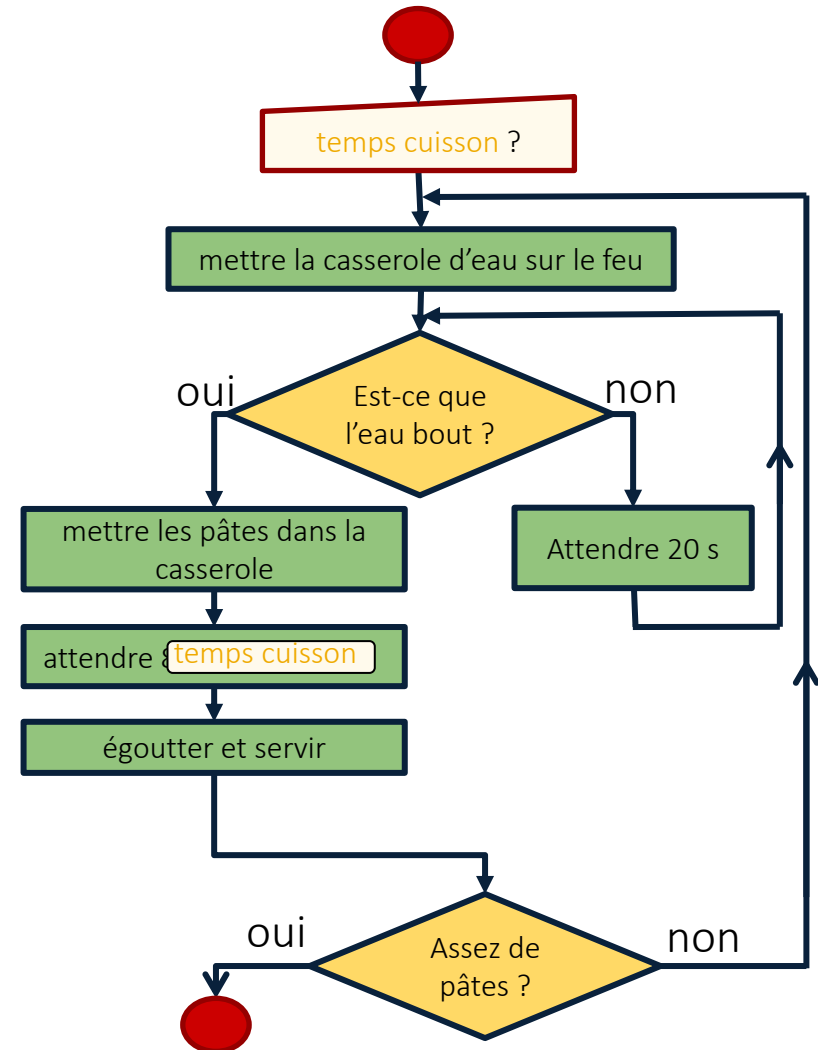
INTRODUCTION AUX VARIABLES

- Est-ce que l'algorithme des pâtes peut s'adapter au goût des personnes en faisant **varier** le **temps de cuisson** ?
- Au début de l'algorithme
 - on crée une variable `temps cuisson`
 - dont on demande la valeur pour l'enregistrer ou la stocker
- et quand on en a besoin
 - on récupère la valeur, avec son nom (étiquette)
 - dans le programme, le nom ou de la variable, remplace sa valeur stockée en mémoire, comme dans une boîte.



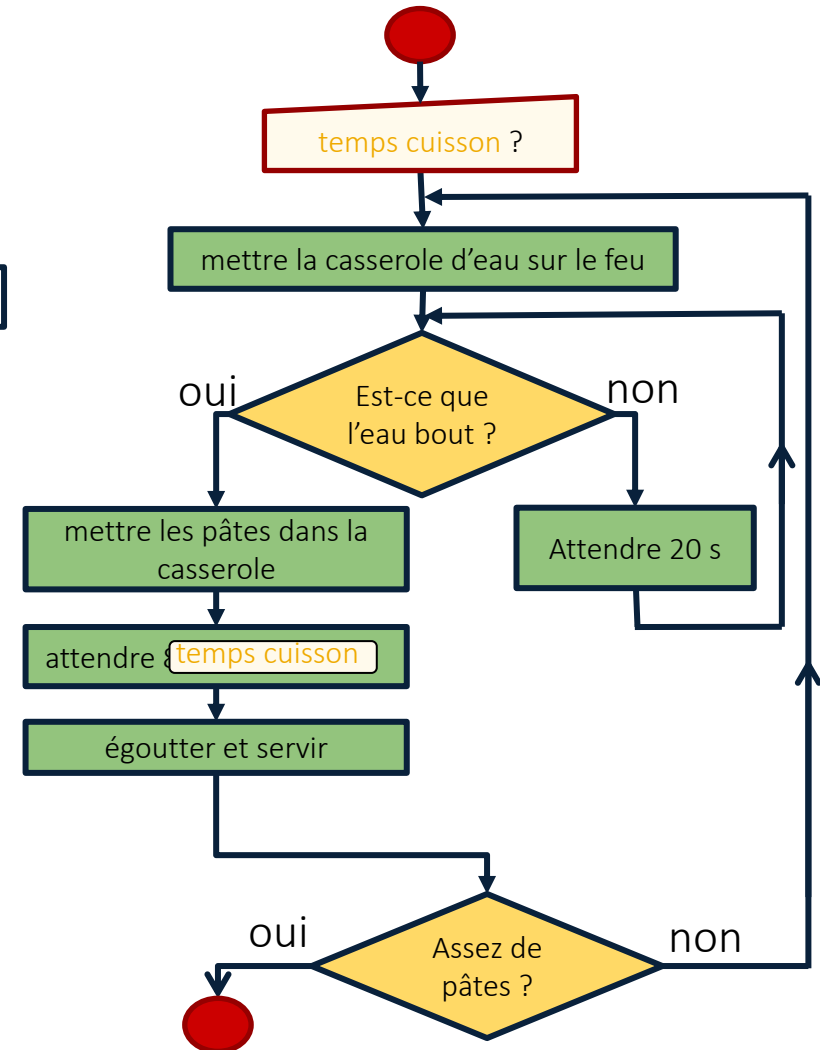
INTRODUCTION AUX PROCÉDURES

- notre algorithme des pâtes peut être exécuté par une personne ... mais pour un robot ?



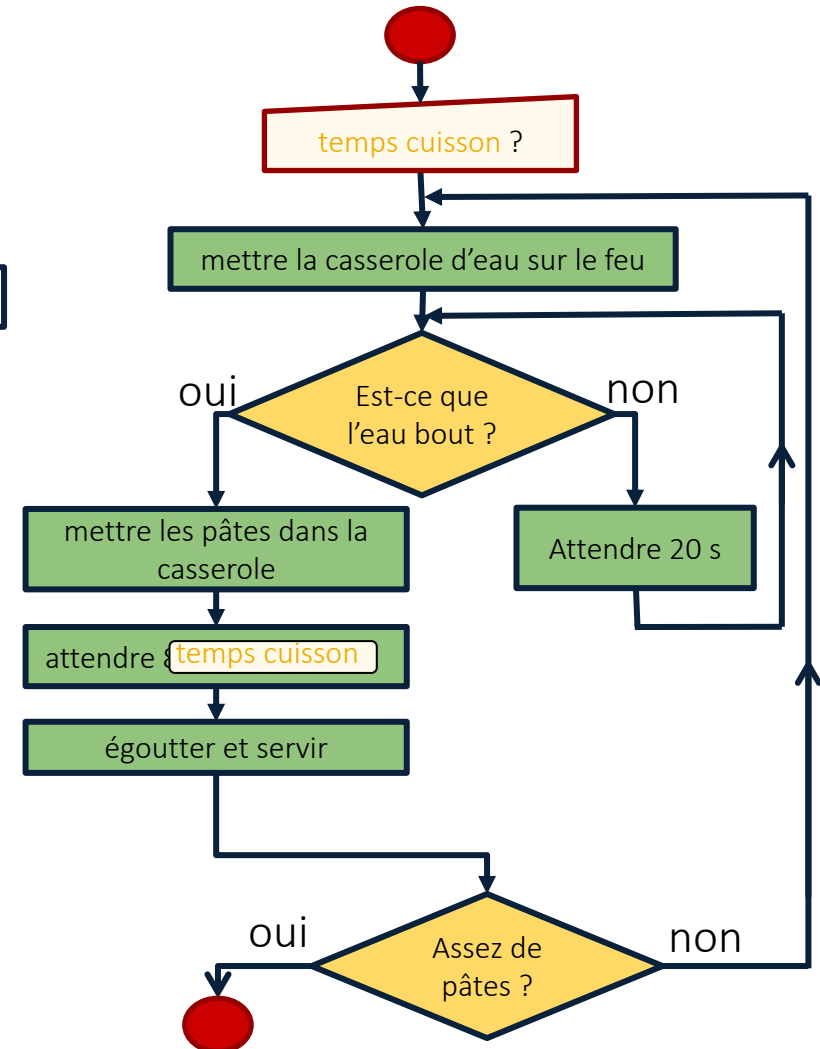
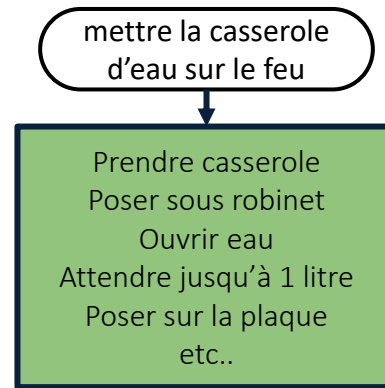
INTRODUCTION AUX PROCÉDURES

- notre algorithme des pâtes peut être exécuté par une personne ... mais pour un robot ?
- Il ne connaît pas la commande mettre la casserole d'eau sur le feu



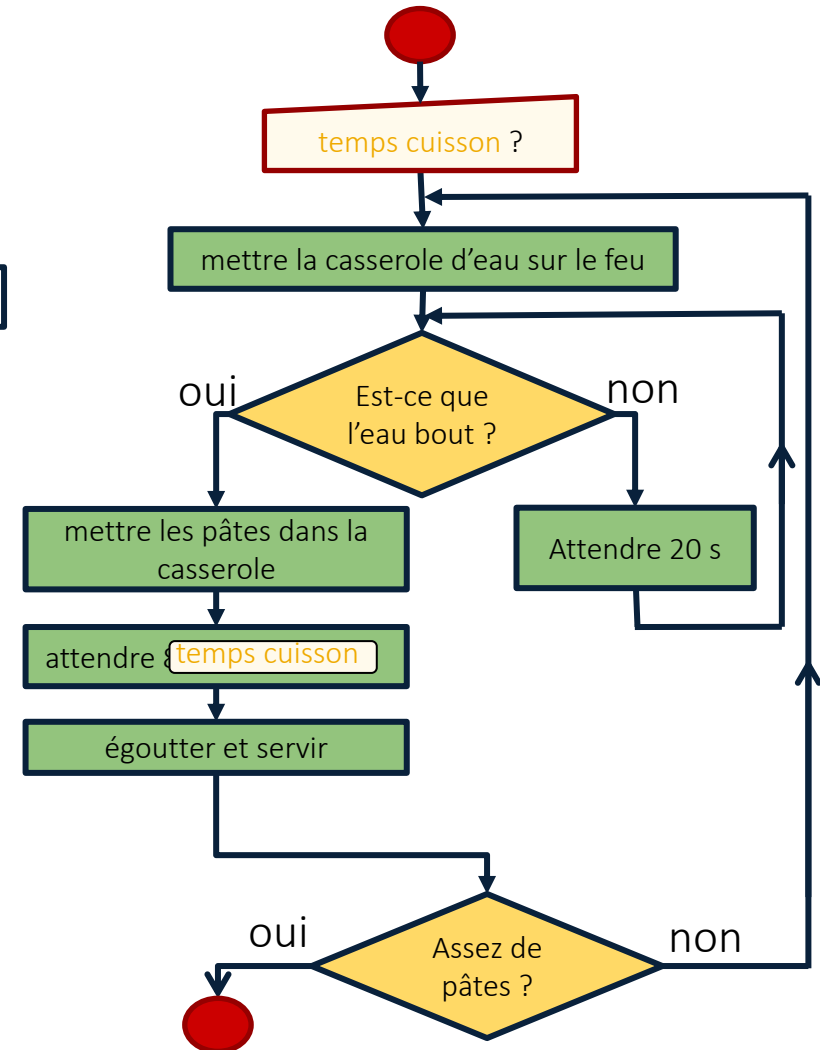
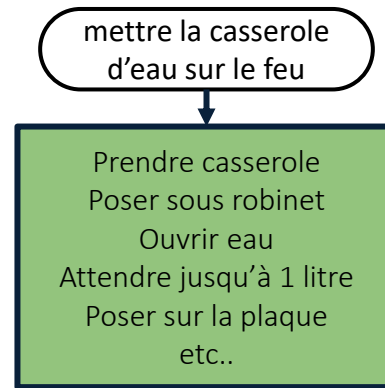
INTRODUCTION AUX PROCÉDURES

- notre algorithme des pâtes peut être exécuté par une personne ... mais pour un robot ?
- Il ne connaît pas la commande mettre la casserole d'eau sur le feu
- on doit donc écrire l'algorithme de cette procédure avec les commandes connues du robot



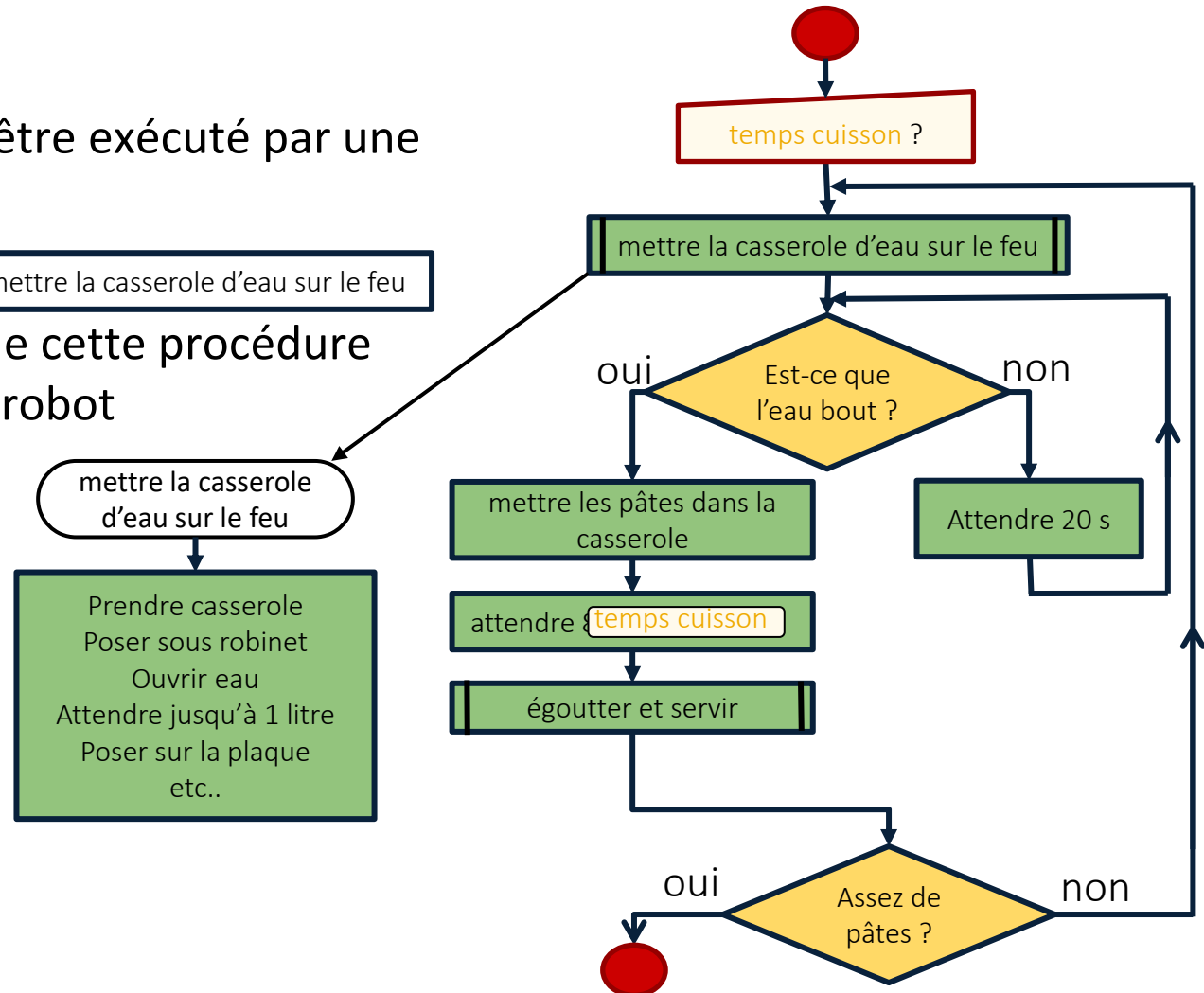
INTRODUCTION AUX PROCÉDURES

- notre algorithme des pâtes peut être exécuté par une personne ... mais pour un robot ?
- Il ne connaît pas la commande mettre la casserole d'eau sur le feu
- on doit donc écrire l'algorithme de cette procédure avec les commandes connues du robot



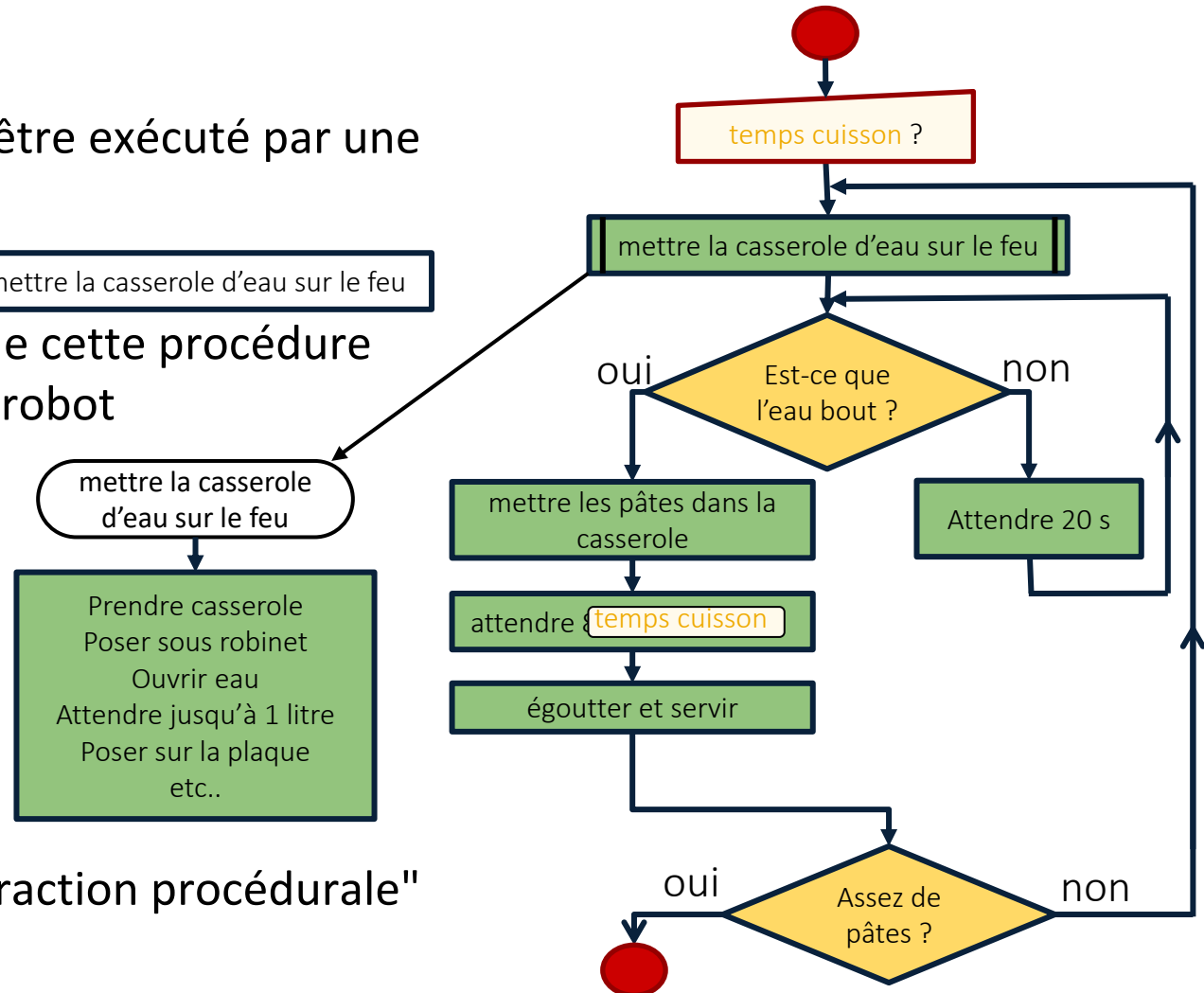
INTRODUCTION AUX PROCÉDURES

- notre algorithme des pâtes peut être exécuté par une personne ... mais pour un robot ?
- Il ne connaît pas la commande `mettre la casserole d'eau sur le feu`
- on doit donc écrire l'algorithme de cette procédure avec les commandes connues du robot
- Puis appeler cette procédure depuis le programme principal (par convention on met des barres doubles en cas d'appel de procédure)



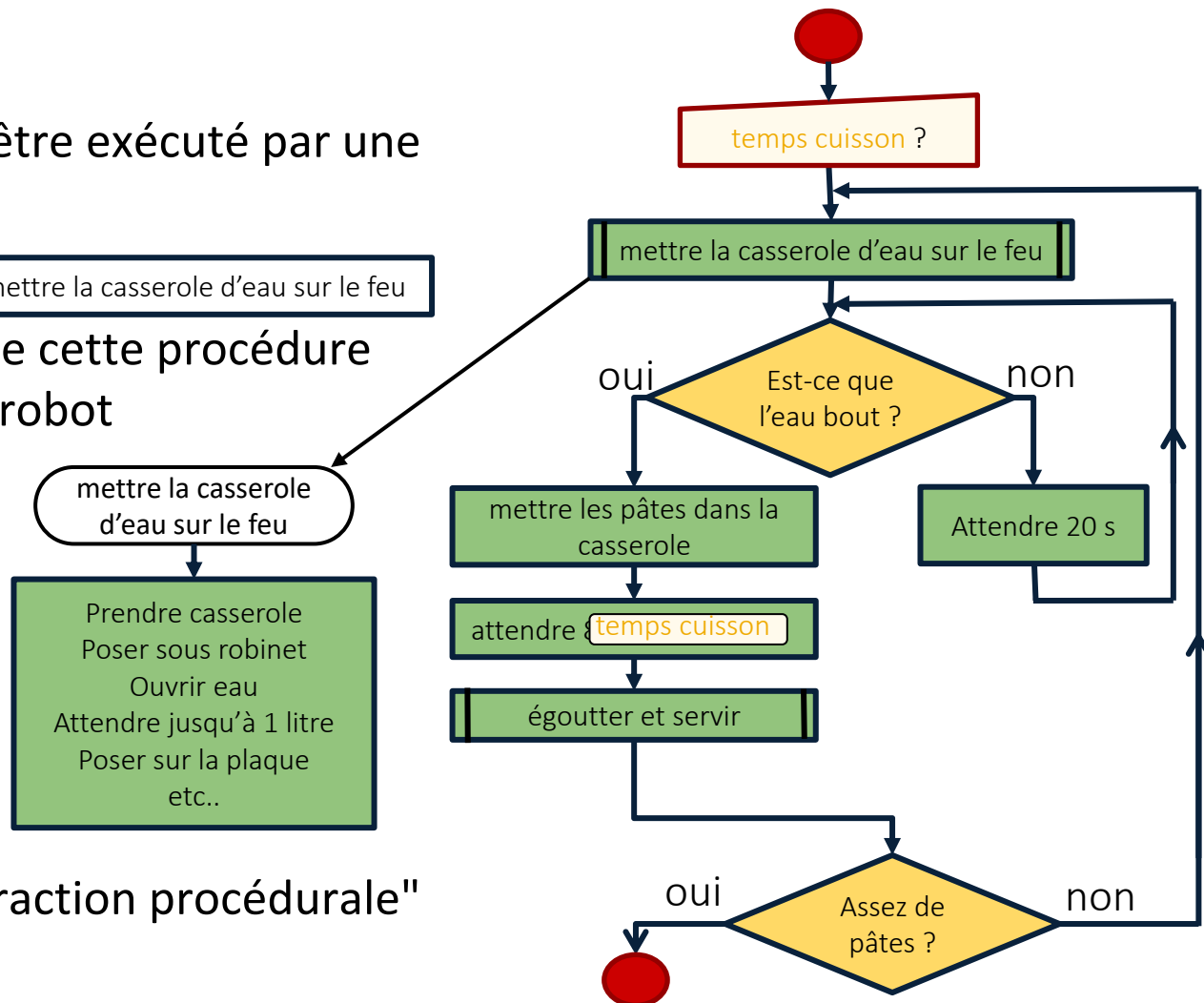
INTRODUCTION AUX PROCÉDURES

- notre algorithme des pâtes peut être exécuté par une personne ... mais pour un robot ?
- Il ne connaît pas la commande `mettre la casserole d'eau sur le feu`
- on doit donc écrire l'algorithme de cette procédure avec les commandes connues du robot
- Puis appeler cette procédure depuis le programme principal (par convention on met des barres doubles en cas d'appel de procédure)
- Avec les procédures on sépare le rôle de la fonction du détail de sa réalisation : c'est une "abstraction procédurale"



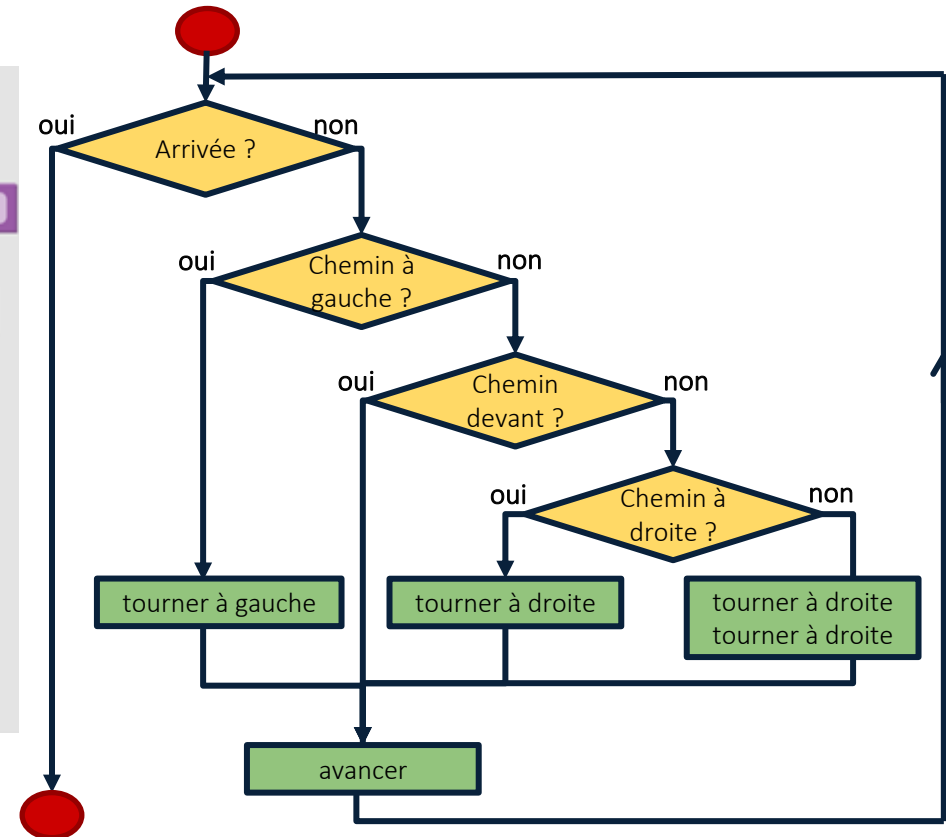
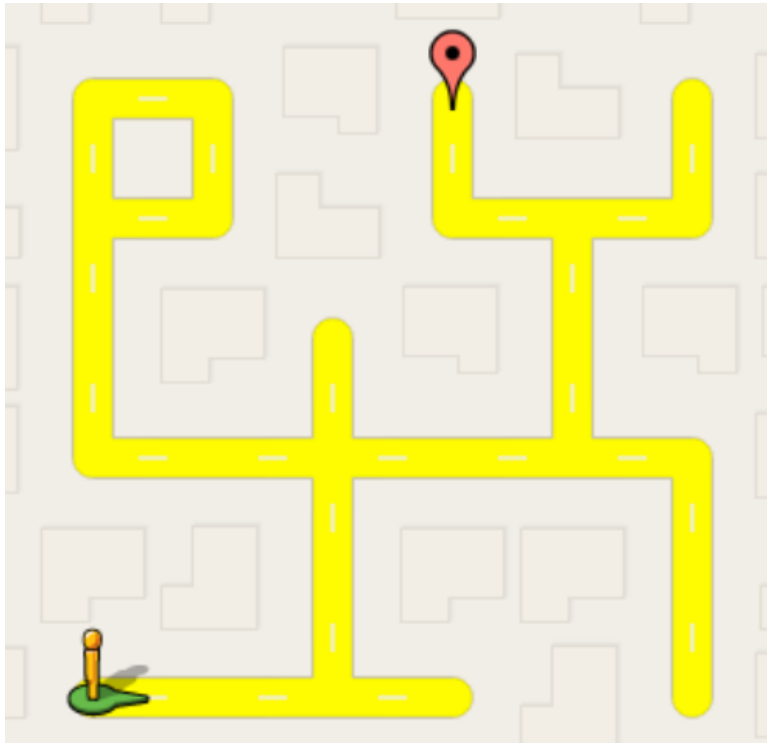
INTRODUCTION AUX PROCÉDURES

- notre algorithme des pâtes peut être exécuté par une personne ... mais pour un robot ?
- Il ne connaît pas la commande `mettre la casserole d'eau sur le feu`
- on doit donc écrire l'algorithme de cette procédure avec les commandes connues du robot
- Puis appeler cette procédure depuis le programme principal (par convention on met des barres doubles en cas d'appel de procédure)
- Avec les procédures on sépare le rôle de la fonction du détail de sa réalisation : c'est une "abstraction procédurale"



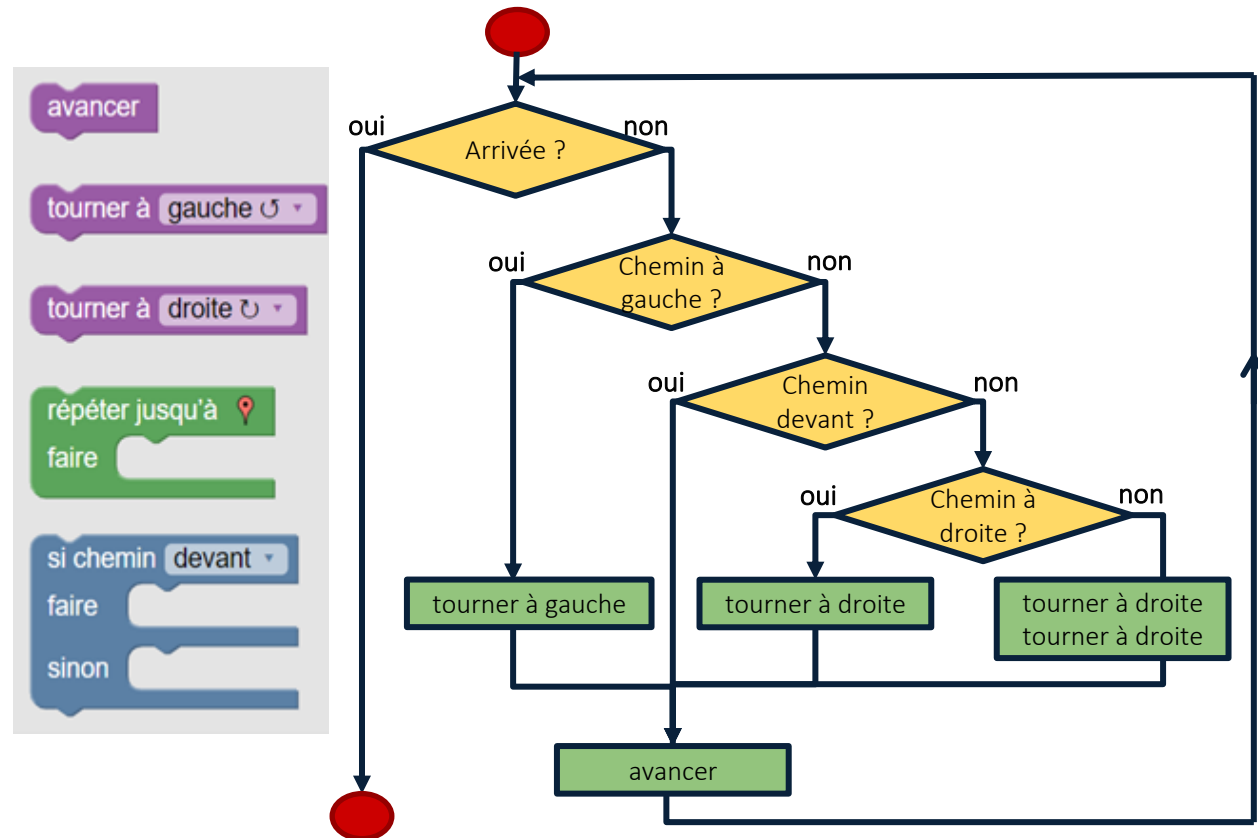
INTRODUCTION AUX PROCÉDURES : EXEMPLE DU LABYRINTHE N10

Blockly maze, niveau 10 : <https://blockly.games/maze?lang=fr&level=10>



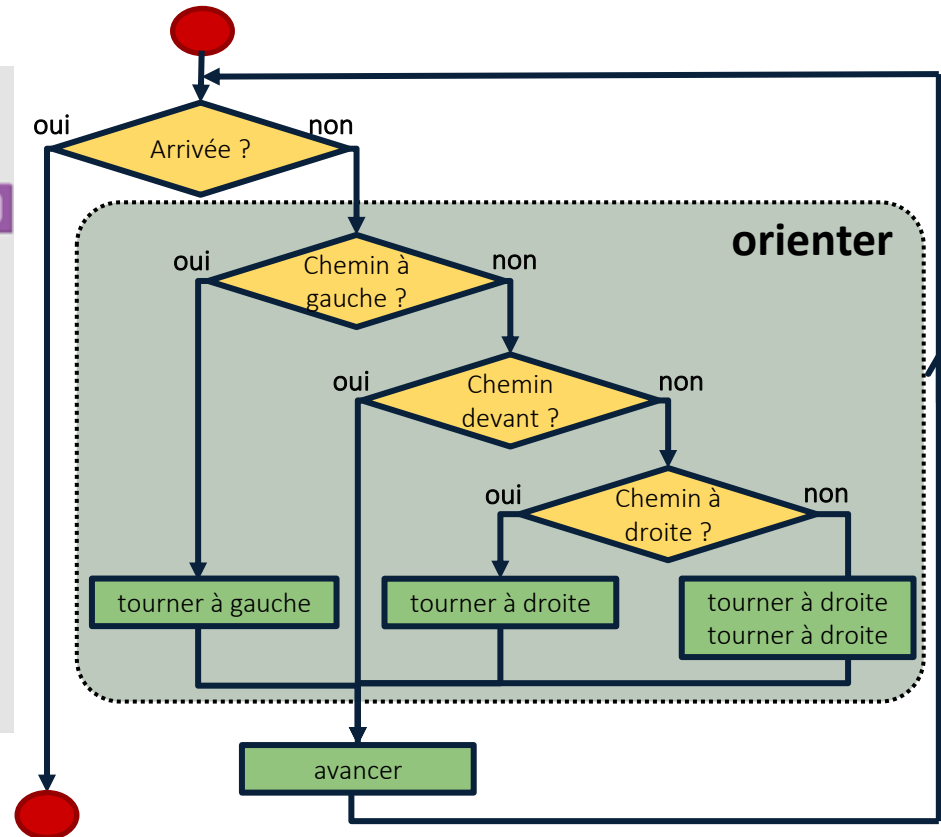
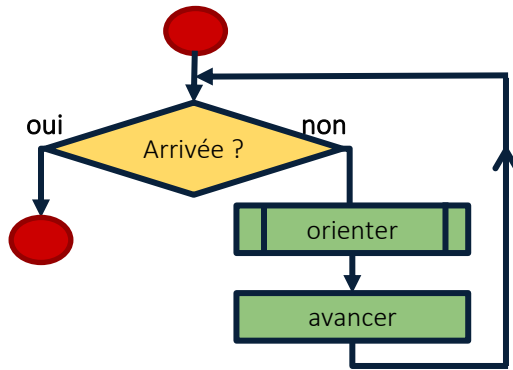
INTRODUCTION AUX PROCÉDURES : EXEMPLE DU LABYRINTHE N10

Blockly maze, niveau 10 : <https://blockly.games/maze?lang=fr&level=10>



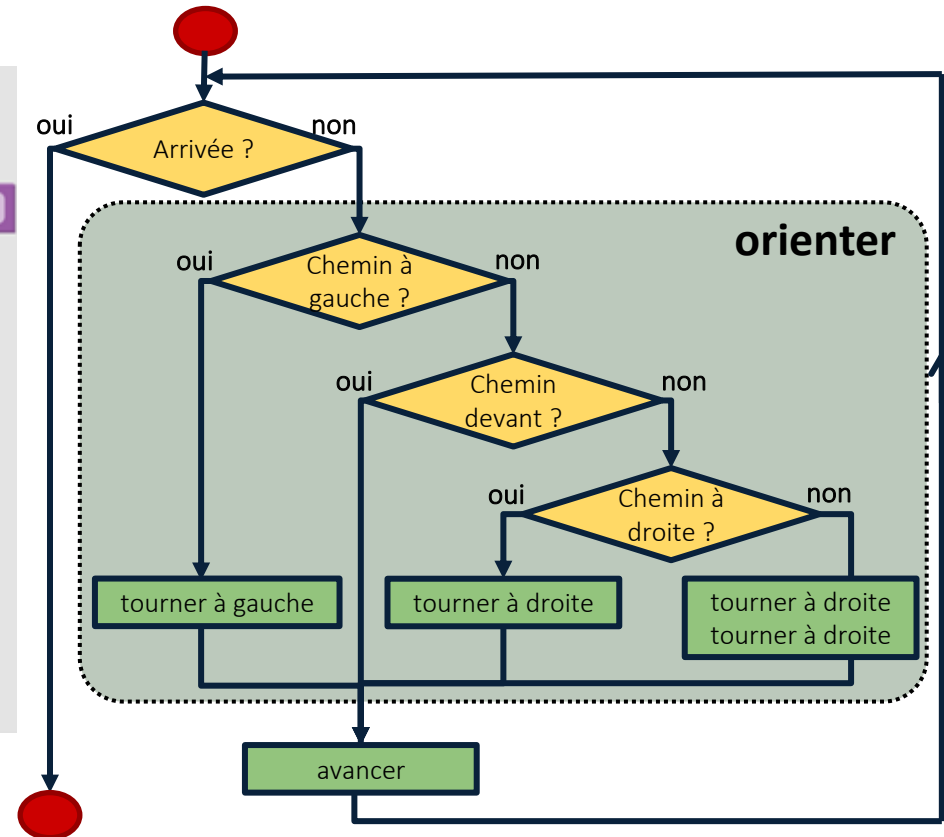
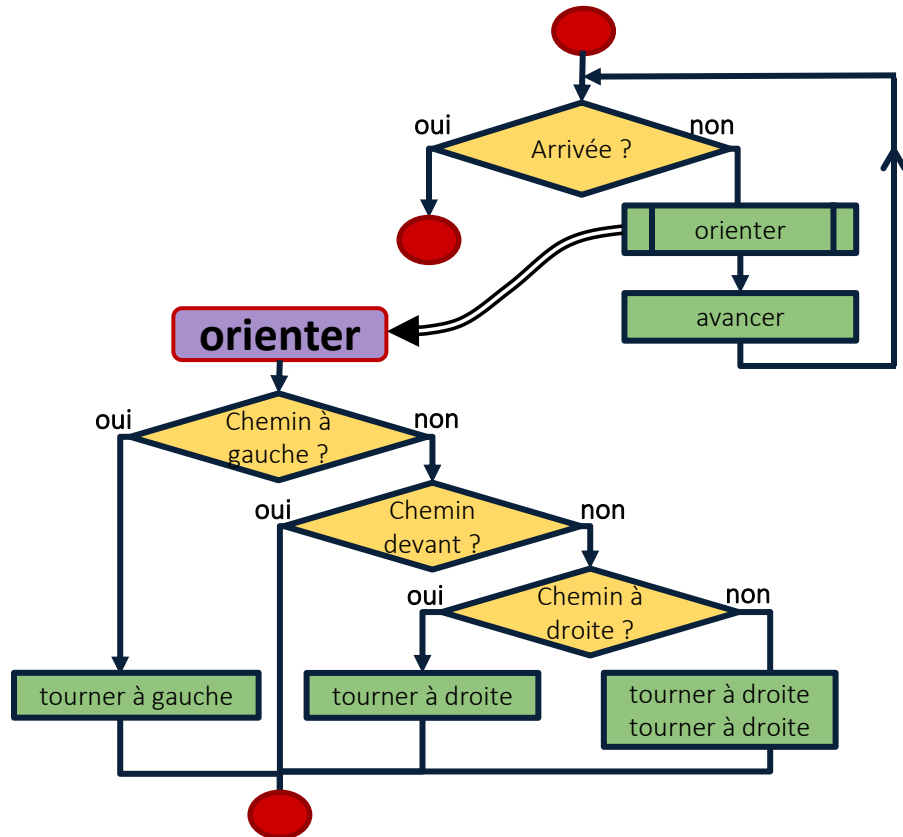
INTRODUCTION AUX PROCÉDURES : EXEMPLE DU LABYRINTHE N10

Blockly maze, niveau 10 : <https://blockly.games/maze?lang=fr&level=10>



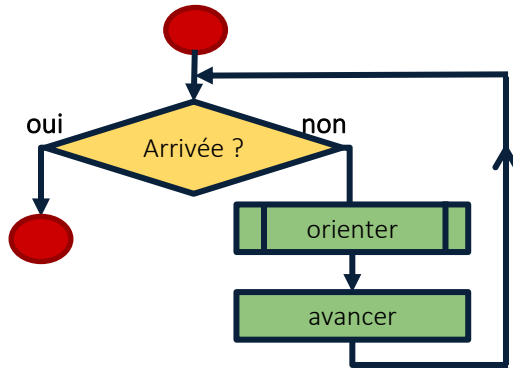
INTRODUCTION AUX PROCÉDURES : EXEMPLE DU LABYRINTHE N10

Blockly maze, niveau 10 : <https://blockly.games/maze?lang=fr&level=10>



INTRODUCTION AUX PROCÉDURES : EXEMPLE DU LABYRINTHE N10

Blockly maze, niveau 10 : <https://blockly.games/maze?lang=fr&level=10>



Pseudo-code

Répéter jusqu'à arrivée
orienter()
avancer



orienter

