

Algorithmique et Programmation

Informatique

- Informatique
 - Information automatique
 - Science du traitement automatique de l'information

- Dualité Matériel-Logiciel
 - Matériel : ordinateurs
 - Logiciels : ensemble de programmes remplissant une fonction déterminée, permettant l'accomplissement d'une tâche informatique donnée

Algorithme et algorithmique

- Algorithme

- mathématicien persan du 9ème siècle Al-Khwarizmi
- description d'une démarche à suivre pour résoudre une série de *tâches* équivalentes en appliquant dans un ordre précis une suite finie d'*opérations*

- Algorithmique

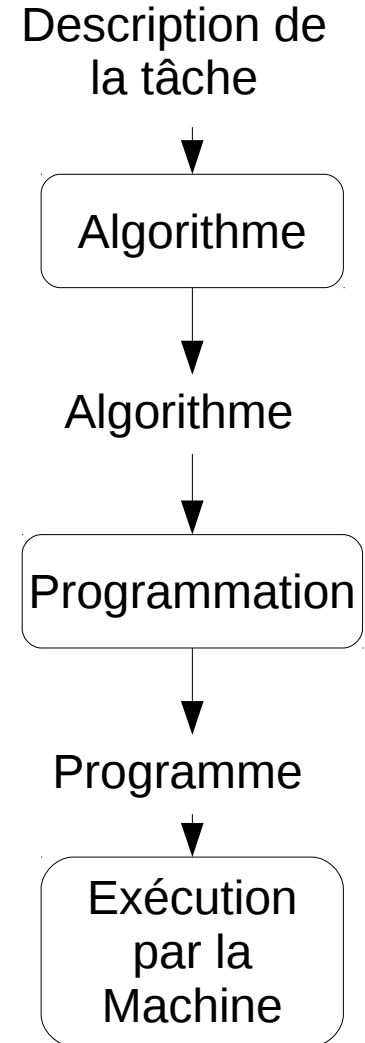
- *art* de construire des algorithmes
 - « Art d'organiser un calcul complexe en partant d'opérations simples », G. Berry
- recherche d'algorithmes ayant de bonnes propriétés
 - **correction**, robustesse, réutilisabilité
 - complexité, efficacité



A postage stamp issued by the USSR in 1983 to commemorate the 1200th anniversary of Muhammad al-Khwarizmi, after whom algorithms are named.

Programmation

- Programmation
 - Traduction de l'algorithme dans la langage de programmation
 - Dépend du langage
 - Étape souvent technique
- Grande variété de langages de programmation
 - **Impératifs**
 - Déclaratifs
 - Fonctionnels
 - Orientés Objets



Les langages de programmation

- « Les langages de programmation, vecteurs de la pensée informatique », G. Berry, Collège de France
 - L'outil d'écriture des programmes
 - Le **support d'interaction avec la pensée**
 - les différents types de langages induisent des styles de pensée : *basés sur des théories du calcul solides*
 - Le support de guerres de religions
- Pourquoi tant de langages
 - Beaucoup d'aspects à traiter ensemble
 - Beaucoup d'innovations successives
 - Énormément de compromis possibles

Objectifs pratiques du cours

- **Fondamentaux d'algorithmique** : être capable de comprendre un algorithme
 - acquérir les concepts fondamentaux d'algorithmique
 - algorithmique impérative uniquement
- **Pratique de l'algorithmique** : être capable de concevoir un petit algorithme
 - formaliser et décomposer une tâche
- **Programmation** : expérimenter l'algorithmique en pratique
 - approfondir la compréhension des algorithmes
 - confrontation la théorie et la pratique

Organisation du cours

- 6 séances

- Introduction à l'algorithmique « générale »
- Introduction à Python 3
- Manipulation des chaînes de caractères
 - Exemple de récupération automatique d'images MODIS
- Utilisation des listes et tableaux
 - Exemple de manipulation d'images
- Utilisation des fonctions
- Notion de programmation orientée objet
 - Exemple des images vectorielles

Documents

- Ressources disponibles en version numérique

http://people.irisa.fr/Thomas.Guyet/enseignements/M1_TELENTVI/

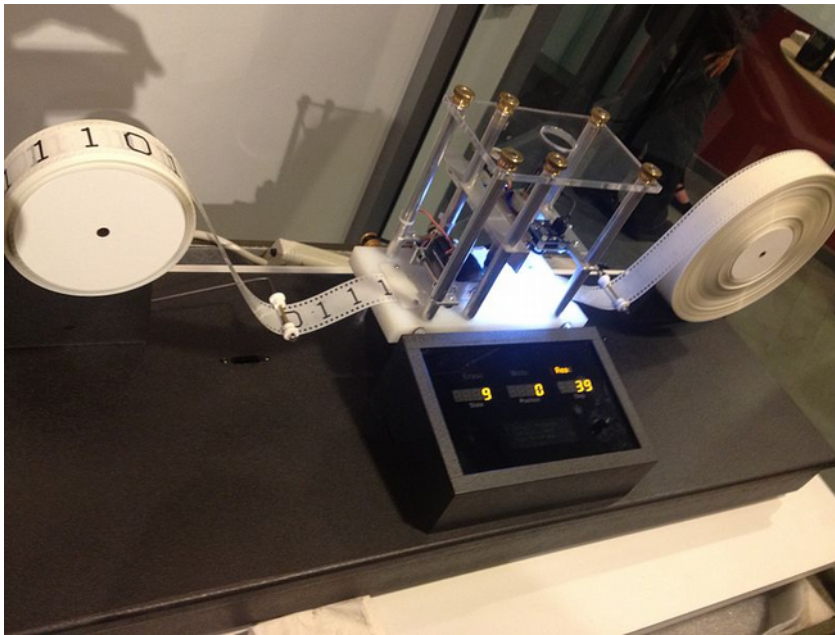
- versions numériques des supports de cours
- Exemples et corrections (partielles) des exercices ...



Qu'est ce que programmer ?

Ordinateur

- *Un ordinateur est une machine à traiter de l'information, de manière formelle, pour autant qu'on lui ait indiqué (dans les détails !) comment mener à bien ce traitement*



Machine de Turing : modèle théorique du fonctionnement d'un ordinateur

Ordinateur = machine formelle

- Les données n'ont pas de « sens » pour une machine

L'ordinateur n'est capable que de *traitements formels* (sur la forme) des informations

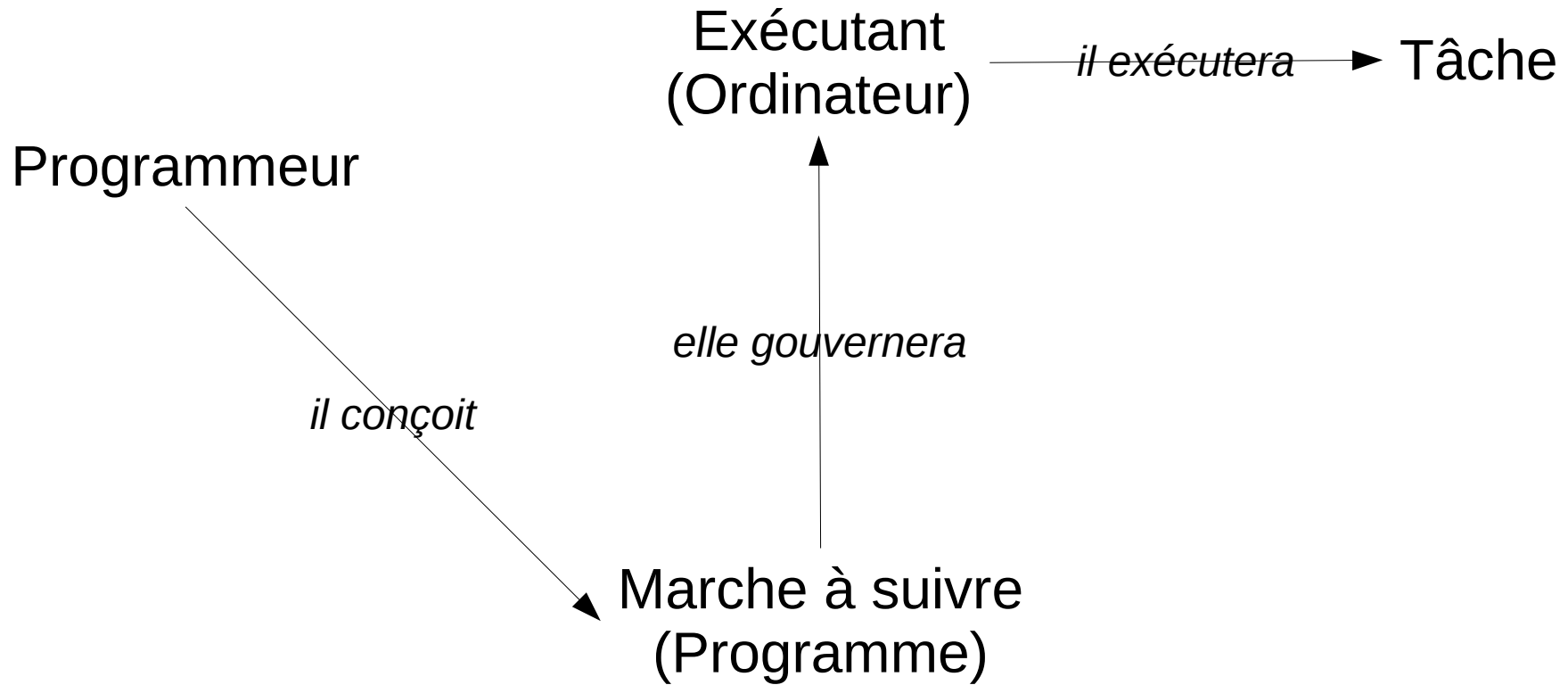
- Exemples de traitement non-formel
 - « résumer un texte »
 - Facile pour un humain, pour une machine ???
 - « écrire un poème »
- Exemples de traitement formel
 - « supprimer un mot sur deux d'un texte »
 - « déterminer le gagnant d'un décathlon »

Ordinateur = exécutant

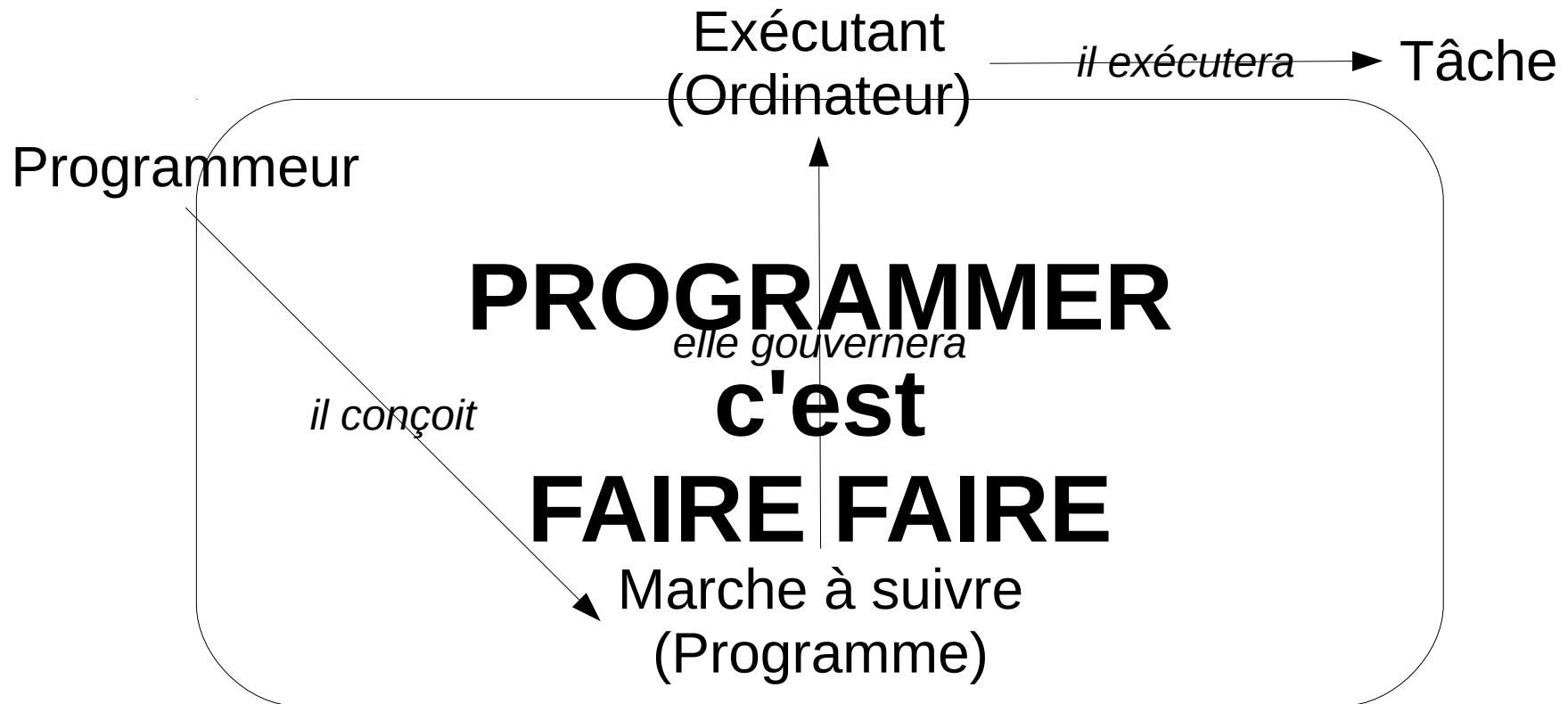
- Un ordinateur ne résout pas de problème !
- **Un ordinateur exécute une marche à suivre correspondant à une tâche**
 - L'accomplissement de la tâche le conduit à produire un résultat qui, pour l'utilisateur, est la solution à un problème

TOUT ce que fait un ordinateur, il le fait gouverné par une « **marche à suivre** », un *programme*

Programmer c'est « faire faire »



Programmer c'est « faire faire »



Les difficultés du « faire faire »

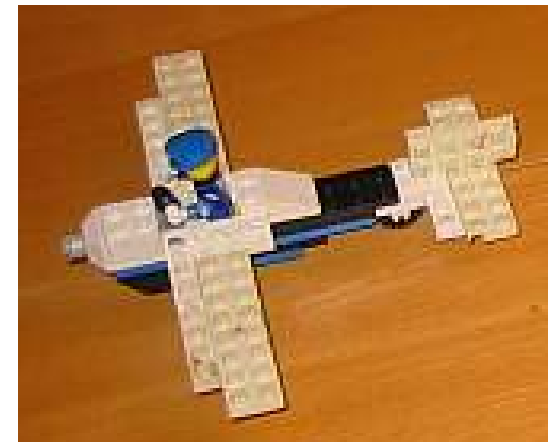
- L'exécutant a des capacités limitées
 - Un ordinateur sait faire un nombre de **tâches élémentaires** limitées
 - *e.g.* comparer deux nombres
 - La tâche qu'on cherche à faire faire est plus **évoluée**
 - *e.g.* trier une série de nombres dans un ordre croissant

NB :

- Certaines tâches évoluées ne peuvent pas être réalisées à partir des tâches élémentaires

Les difficultés du « faire faire »

- L'exécutant a des capacités limitées
 - La tâche à réaliser peut paraître simple au programmeur, mais sa description en vue de sa réalisation par un ordinateur n'en est pas moins difficile
 - Programmer = **décortiquer** et **expliquer la réalisation d'une tâche complexe à partir d'un « vocabulaire de tâches élémentaires »** que sait faire un ordinateur
 - **décortiquer** : c'est un problème de « Lego » : comment faire un réalisation ressemblante avec un nombre de limité de forme de base



Les difficultés du « faire faire »

- On « fait faire » **en différé**

- Il faut fournir la marche à suivre avant l'exécution, par l'ordinateur, de celle-ci.

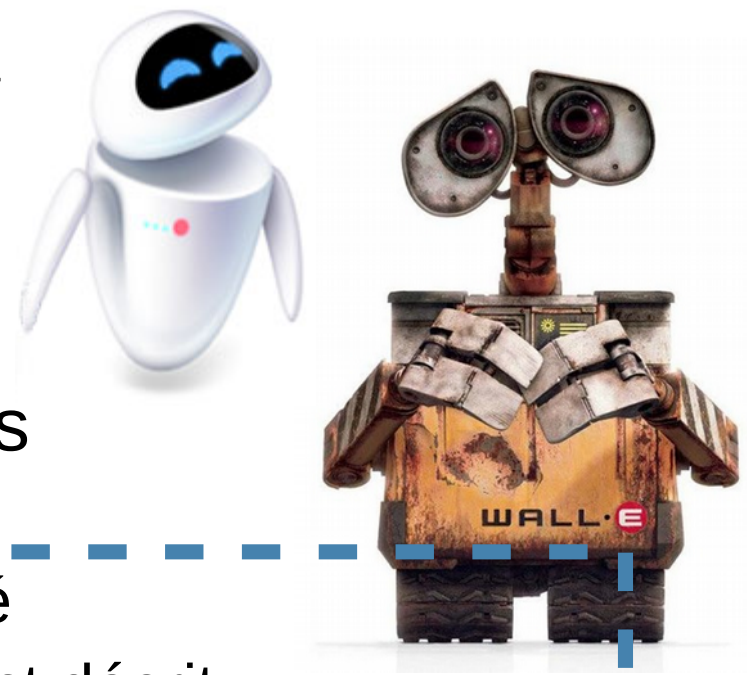
- **TOUT** doit être prévu à l'avance dans la marche à suivre

- **non-ambiguë** : rien de doit être laissé à l'appréciation de l'exécutant

- **complète** : tous les cas possibles doivent être prévus



Les difficultés du « faire faire »



- L'ordinateur est un robot
 - Ce n'est pas un humain à qui vous expliquez quelque chose
 - **Exhaustivité** : tout doit être explicité
 - **Précision** : tout doit être précisément décrit
 - **Formalisation** : tout doit être écrit avec cette rigueur
 - Ces contraintes de rigueur sont proprement « *inhumaines* » !
 - Contraintes nécessaires, sinon l'étape de programmation sera très fastidieuse
 - Le robot vous apparaîtra rapidement comme « borné » et « bête » !

Les difficultés du « faire faire »

- Pour résumer les difficultés
 - l'exécutant à des capacités limités
 - on fait faire en différé
 - l'ordinateur est un robot
- Au final, programmer c'est

rédiger une *marche à suivre* **complète** pour FAIRE
FAIRE une tâche par un exécutant aux **capacités limités**

Fondamentaux algorithmiques

Quelques exemples courants d'algorithmes

The image displays four staves of musical notation in treble clef, illustrating various performance algorithms. The first staff is marked **Andante rubato** and begins with a *p* dynamic. The second staff starts at measure 5. The third staff, starting at measure 9, includes markings for *accel.* (accelerando), *rall.* (ritardando), and *dim.* (diminuendo). The fourth staff, starting at measure 13, is marked **Poco più mosso, molto appassionato** and includes markings for *molto rubato*, *p dolcissimo*, and *molto sostenuto*.

Quelques exemples courants d'algorithmes

■ SIGNATURE

Vous pouvez enregistrer une signature à ajouter automatiquement à vos SMS. La signature n'est pas affichée lors de la saisie du texte du message, mais elle apparaît à la réception du message par le destinataire. La taille maximum de la signature est de 30 caractères. Si le texte du message envoyé atteint 918 caractères, la signature ne peut être ajoutée.

Pour enregistrer une signature :

- ▶ 1 Appuyez sur **OK**. Sélectionnez **SMS**.
- 2 Sélectionnez **Paramètres**.
- 3 Sélectionnez **Signature**.
- 4 Sélectionnez **Insérer au message**. Sélectionnez **Oui**.
- 5 Sélectionnez **Texte** et saisissez le texte de votre signature.
- 6 Sélectionnez **Enreg.** pour enregistrer votre signature.

■ EMISSION DE SMS A PARTIR DE L'ARCHIVE SIM ET DE LA BOÎTE D'ENVOI

La Boîte d'envoi et l'Archive SIM contiennent des brouillons de messages (non envoyés) et des messages envoyés enregistrés (délivrés ou non délivrés). Ces messages peuvent être sélectionnés à partir du menu de la boîte d'envoi ou de l'archive SIM et peuvent être modifiés et envoyés à nouveau comme SMS.

Pour sélectionner l'un de ces messages :

- ▶ 1 Appuyez sur **OK**. Sélectionnez **SMS**.
- 2 Sélectionnez **Boîte d'envoi** ou **Archive SIM** puis **SMS émis** et utilisez les touches de navigation Haut et Bas pour faire défiler la liste jusqu'au message voulu. Les messages sont « transmis » ou « à envoyer ».
- 3 Appuyez sur **Options** pour choisir **Voir**, **Supprimer**, **Supprimer tout**, **Etat** (des messages envoyés seulement) **Envoyer**, **Envoi e-mail**, **Modifier**, **Déplacer vers SIM** (pour les messages de la boîte d'envoi seulement), **Numéros du SMS** (pour enregistrer ou appeler des numéros contenus dans l'en-tête ou le texte du SMS) ou **Détails**.

Quelques exemples courants d'algorithmes



Départ : Rennes, Place de la Gare

1 **D177** Continuer sur : D177 / Place de la Gare

2 **Entrer dans Rennes** 3 km

- ↩ Prendre à gauche : Avenue Jean Janvier (0.4 km)
- ↩ Prendre à gauche : Boulevard de la Liberté (0.8 km)
- Continuer sur : D125 / Place de Bretagne (0.4 km)
- Continuer sur : D125 (1.5 km)

3 km 00h07



Arrivée : Rennes, 65 Rue de Saint-Brieuc

3 km 00h07

Quelques exemples courants d'algorithmes

Préparation : 25 min
Cuisson : 40 min



Ingédients (pour 4 personnes) :

- 1 rouleau de pâte feuilletée
- 5 endives
- 2 crottins de Chavignol
- 50 g de beurre
- 1 cuillère à café d'herbes de Provence
- 1 cuillère à soupe de sucre
- sel et poivre

Préparation :

Coupez en quatre les endives, dans le sens de la longueur.

Les disposer dans une sauteuse et les faire revenir dans la moitié du beurre, à feu doux durant 25 min; elles doivent prendre une belle coloration blonde.

Dans un moule à manqué, versez l'autre moitié du beurre fondu et saupoudrez de la cuillère à soupe de sucre.

Préchauffez le four à 200°C (thermostat 6-7).

Disposez les quartiers d'endives en rosace, au fond du moule (le coeur au centre).

Coupez les crottins de chèvre en tranches fines et disposez-les sur les endives, puis saupoudrez de la moitié de la cuillère d'herbes de provence; salez et poivrez.

Recouvrez avec la pâte feuilletée, le bord de la pâte doit glisser entre le moule et les endives.

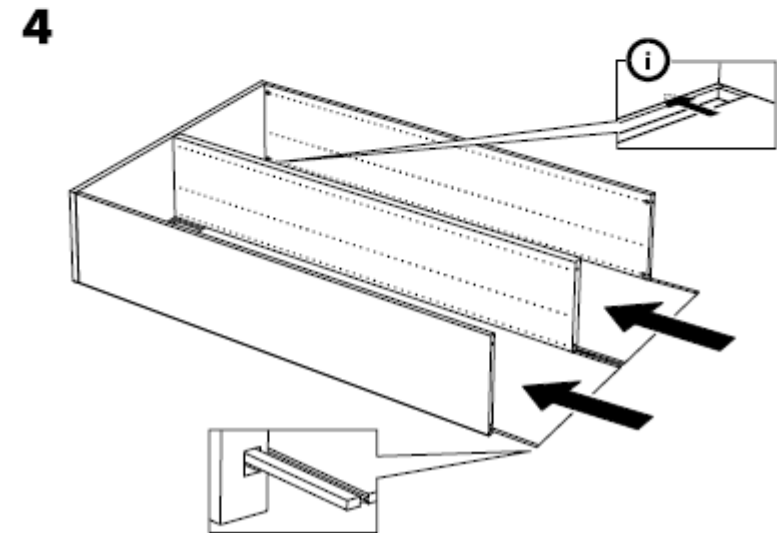
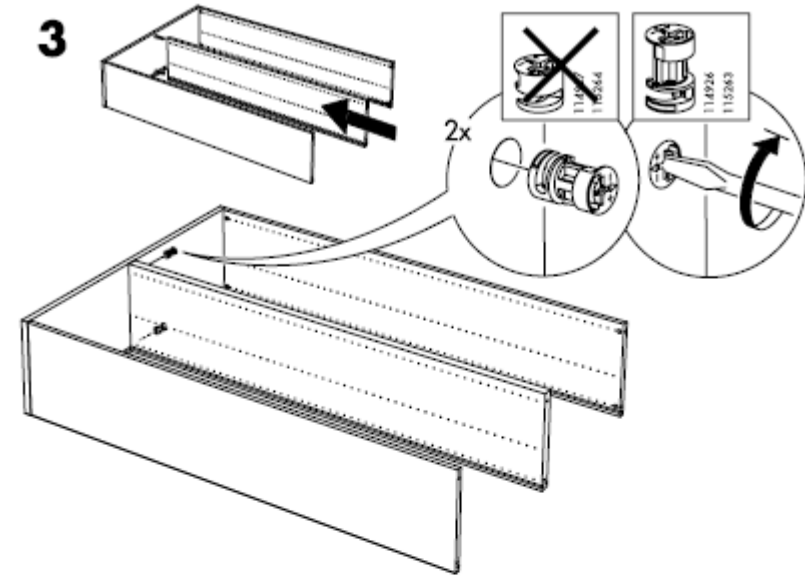
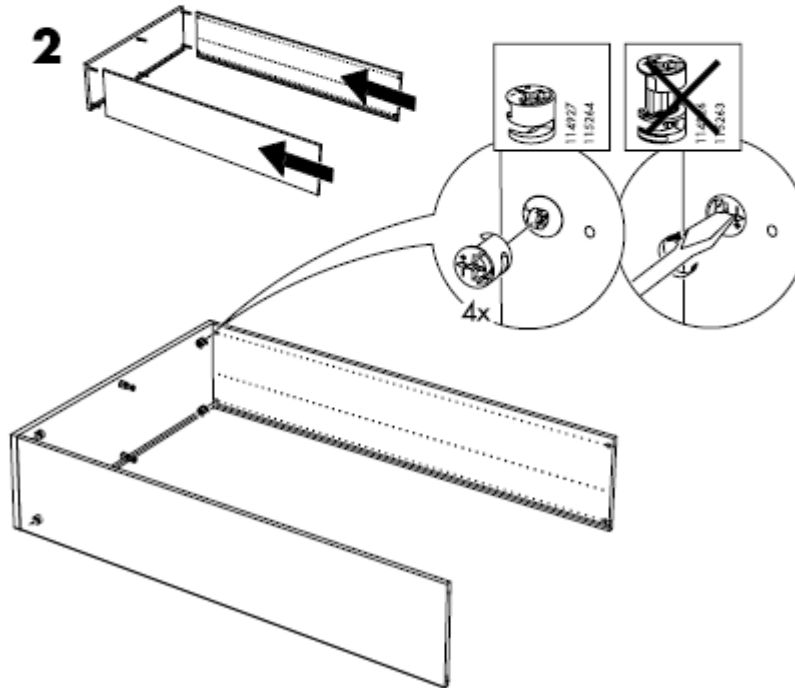
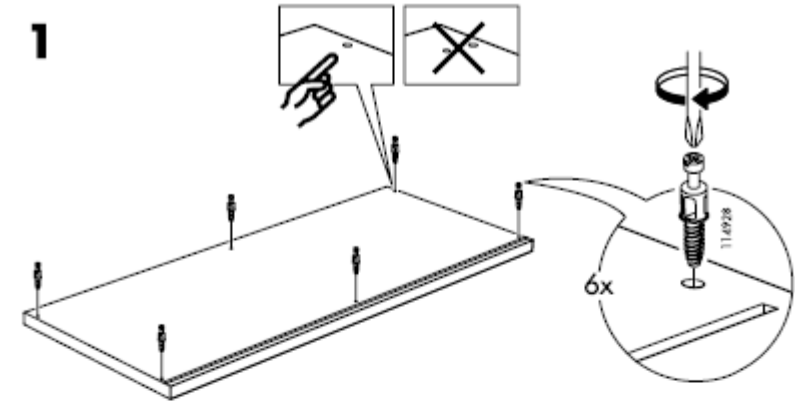
Entourez 20 min, la pâte doit se soulever et prendre une belle couleur blonde.

Prenez une assiette légèrement plus grande que le diamètre du moule, puis posez-la dessus, et d'un geste retournez l'ensemble.

Attendez 5 ou 6 secondes, et soulevez délicatement le moule.

Pour finir, saupoudrez avec la moitié des herbes de Provence restantes.

Quelques exemples courants d'algorithmes



Constituants d'un algorithme

- 1) Instructions
- 2) Structures de contrôles
 - 1) Séquence
 - 2) Alternative
 - 3) La répétition
 - 4) Le branchement
- 3) Approche descendante : les procédures

Les instructions

Les devises Shadok

- Exemples :

- Appuyer sur la touche OK
- Sélectionner Paramètres
- Tourner à droite



IL VAUT MIEUX POMPER MÊME S'IL NE SE PASSE RIEN QUE RISQUER QU'IL SE PASSE QUELQUE CHOSE DE PIÈRE EN NE POMPANT PAS.

- Il s'agit d'**instructions d'action élémentaire**

- action élémentaire : *une action dont l'exécutant est capable*
- Ordonne la réalisation de l'action élémentaire de l'exécutant
 - On donne l'instruction, il la fait !
 - On parle de programmation « **impérative** »

La séquence

- C'est une suite d'instructions

- 2 Sélectionnez **Paramètres**.
- 3 Sélectionnez **Signature**.
- 4 Sélectionnez **Insérer au message**.

L'exécutant,
sélectionne le menu *Paramètres*, **puis**
sélectionne le menu *Signature*, **puis**
Sélectionne le menu *Insérer au message*

- Représentation sous forme de pseudo-code

```
...  
Fais ceci  
Fais cela  
...
```

L'alternative

- Permet de prévoir des comportements différents en fonction des **conditions d'exécutions**
- Pseudo-code

```
SI condition ALORS
└   A
suite
```

```
SI condition ALORS
└   A
SINON
└   B
suite
```

- Le **SINON** est optionnel dans la structure
- A et B sont deux « **morceaux d'algorithmes** »
 - peuvent contenir une séquence de plusieurs instructions !
 - aussi appelé « *bloc d'instructions* »
- La **condition** doit contenir une **condition !!!!**

Indentation : le repérage des blocs d'instructions

- Indentation = décalage du texte vers la droite
- L'indentation visualise les blocs d'instructions concernés par une structure de contrôle
- La barre verticale met en évidence l'indentation

```
SI condition ALORS  
|   A  
SINON  
|   B  
suite
```

Exemple : beurrer une tartine

```
Prendre le beurre  
SI plus de pain ALORS  
|   Prendre la voiture  
|   Aller à la boulangerie  
|   Acheter du pain  
Trancher le pain  
Beurrer
```

Conditions

- Une condition ou (*test de condition*) est une **assertion qui peut être testée** par la machine
- Une condition ne peut prendre que deux valeurs :
 - Si la condition est *vérifiée* : la condition est **vraie**
 - Sinon, la condition est **fausse**
- Les conditions peuvent être combinées à l'aide d'opérateurs logiques
 - AND, OR, NOT

L'alternative

- Exemple :
 - Condition : « température inférieure à 4°C »
 - Assertion qui est soit vraie, soit fausse !

```
SI température inférieure à 4°C ALORS
```

```
    Prendre un manteau
```

```
    Prendre des gants
```

```
SINON
```

```
    Prendre une veste
```

```
Sortir en ville
```


L'alternative

- Exemple :
 - Condition : « température inférieure à 4°C »
 - Assertion qui est soit vraie, soit fausse !

est vrai ...

```
SI température inférieure à 4°C ALORS
```

```
    Prendre un manteau
```

```
    Prendre des gants
```

```
SINON
```

```
    Prendre une veste
```

```
Sortir en ville
```

L'alternative

- Exemple de combinaison de conditions

```
SI temp. supérieure à 4°C et il pleut ALORS  
└ Prendre un K-Way  
SI temp. inférieure à 0°C ou il pleut ALORS  
└ Prendre une veste  
Sortir en ville
```

Que se passe-t-il si il pleut et qu'il fait 10°C ?

Que se passe-t-il si il ne pleut pas et qu'il fait 12°C ?

Que se passe-t-il si il pleut et qu'il fait 2°C ?

Que se passe-t-il si il ne pleut pas et qu'il fait 2°C ?

Y'a-t-il un risque que je prenne un K-way et une veste ?

L'alternative

- Exemple : séparation de « cas »
 - SI ... SINON SI ... SINON

```
SI température inférieure à 4°C ALORS
```

```
    Prendre un manteau
```

```
    Mettre des gants
```

```
SINON SI température inférieure à 18°C ALORS
```

```
    Prendre une veste
```

```
SINON SI température supérieure à 22°C ALORS
```

```
    Enlever un pull
```

```
    Mettre un chapeau
```

```
SINON
```

```
    Enlever un pull
```

Que se passe-t-il si il fait 3°C ?

Que se passe-t-il si il fait 12°C ?

Que se passe-t-il si il fait 20°C ?

L'alternative

- Exemple : séparation de « cas »
 - Attention à l'ordre des conditions ...

```
SI température inférieure à 18°C ALORS
|   Prendre une veste
SINON SI température inférieure à 4°C ALORS
|   Prendre un manteau
|   Mettre des gants
SINON SI température supérieure à 22°C ALORS
|   Enlever un pull
|   Mettre un chapeau
SINON
|   Enlever un pull
```

Que se passe-t-il si il fait 3°C ?

L'alternative

- Exemple d'imbrication de structures conditionnelles

SI Il pleut **ALORS**

 Mettre des bottes

SI température inférieure à 10°C **ALORS**

 Prendre un Kway

SINON

 Prendre un parapluie

SINON

SI température inférieure à 4°C **ALORS**

 Prendre un manteau

SINON SI température inférieure à 18°C **ALORS**

 Prendre une veste

SINON

 Ne rien prendre

L'alternative

- Exemple d'imbrication

```
SI Il pleut ALORS
```

```
  Mettre des bottes
```

```
    SI température inférieure à 10°C ALORS
```

```
      Prendre un Kway
```

```
    SINON
```

```
      Prendre un parapluie
```

```
SINON
```

```
  SI température inférieure à 4°C ALORS
```

```
    Prendre un manteau
```

```
  SINON SI température inférieure à 18°C ALORS
```

```
    Prendre une veste
```

```
  SINON
```

```
    Ne rien prendre
```

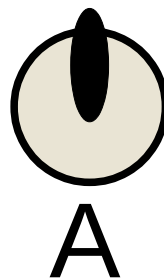
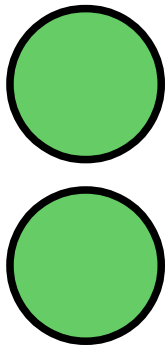
Ce bloc peut être supprimé !

Importance de l'indentation

- Les instructions :

```
SI le voyant du haut est vert ALORS  
  Tourner sur la manette A à droite  
SI le voyant du bas est vert ALORS  
  Tourner sur la manette B à droite  
SINON  
  Tourner les deux manettes à gauche
```

- Que dois-je faire dans la situation suivante ?

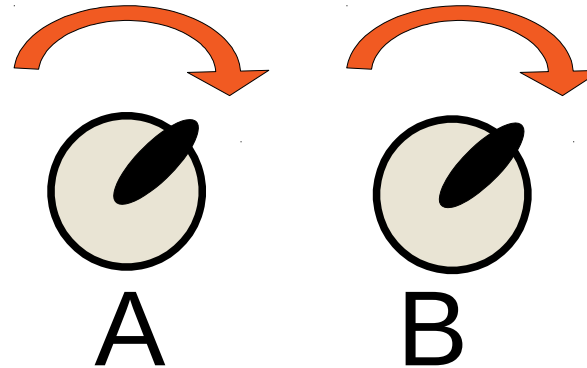
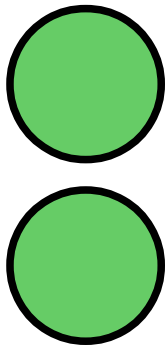


Importance de l'indentation

- Les instructions :

```
SI le voyant du haut est vert ALORS  
  Tourner sur la manette A à droite  
SI le voyant du bas est vert ALORS  
  Tourner sur la manette B à droite  
SINON  
  Tourner les deux manettes à gauche
```

- Que dois-je faire dans la situation suivante ?

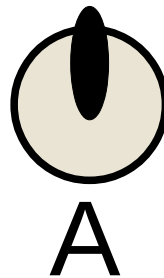
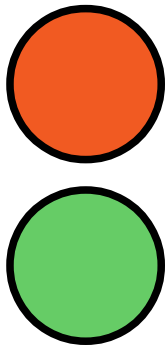


Importance de l'indentation

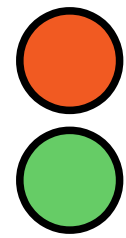
- Les instructions :

```
SI le voyant du haut est vert ALORS  
  Tourner sur la manette A à droite  
SI le voyant du bas est vert ALORS  
  Tourner sur la manette B à droite  
SINON  
  Tourner les deux manettes à gauche
```

- Que dois-je faire dans la situation suivante ?



Importance de l'indentation

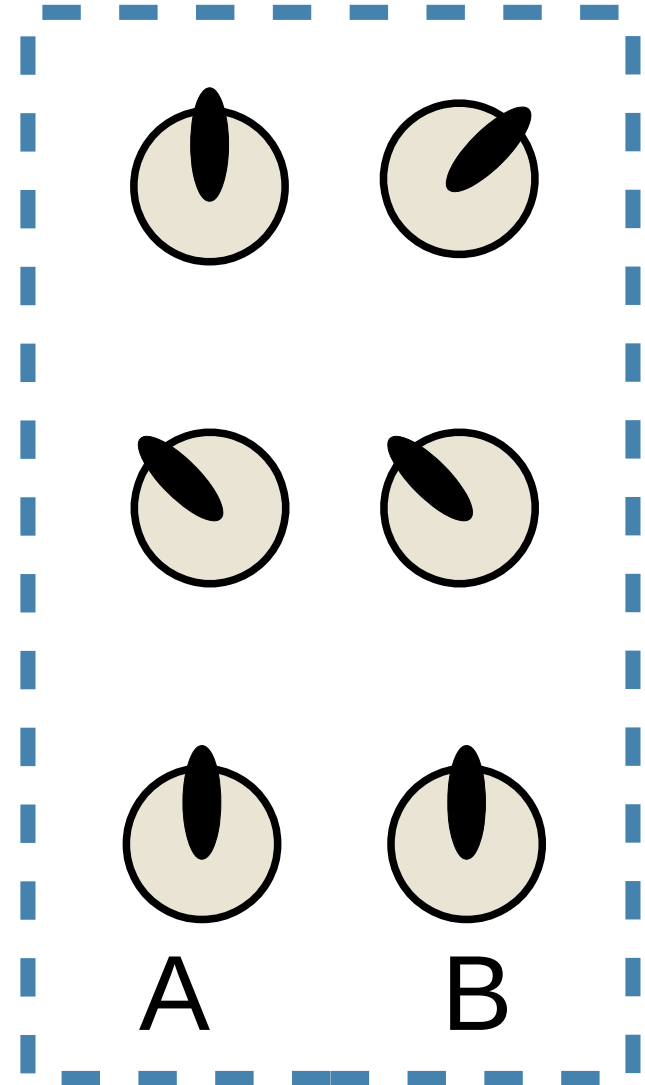


- Trois solutions en fonction de l'indentation

```
SI le voyant du haut est vert ALORS
├── Tourner sur la manette A à droite
SI le voyant du bas est vert ALORS
├── Tourner sur la manette B à droite
SINON
├── Tourner les deux manettes à gauche
```

```
SI le voyant du haut est vert ALORS
├── Tourner sur la manette A à droite
│   └── SI le voyant du bas est vert ALORS
│       └── Tourner sur la manette B à droite
SINON
├── Tourner les deux manettes à gauche
```

```
SI le voyant du haut est vert ALORS
├── Tourner sur la manette A à droite
│   └── SI le voyant du bas est vert ALORS
│       └── Tourner sur la manette B à droite
SINON
├── Tourner les deux manettes à gauche
```



La répétition

- Deux formes génériques
 - **Répéter** un groupe d'instructions **tant qu'une condition est vérifiée**
 - **Répéter** un groupe d'instructions **jusqu'à ce qu'une condition soit vérifiée**

```
TANT QUE condition FAIRE
  |
  | ...
  |
suite
```

```
REPETER
  |
  | ...
  |
JUSQU'À CE QUE condition
suite
```

La répétition

Mettre les pâtes à cuire

TANT QUE des pâtes ne sont pas *al dente* **FAIRE**

 mélanger les pâtes

 laisser cuire les pâtes 2 minutes

Servir

- Lors de l'exécution

Mettre les pâtes à cuire

Vérifier la cuisson (cuisson = pas al dente)

Mélanger les pâtes

Laisser cuire 2 min

Vérifier la cuisson (cuisson = pas al dente)

Mélanger les pâtes

Laisser cuire 2 min

Vérifier la cuisson (cuisson = pas al dente)

Mélanger les pâtes

Laisser cuire 2 min

Vérifier la cuisson (cuisson = al dente)

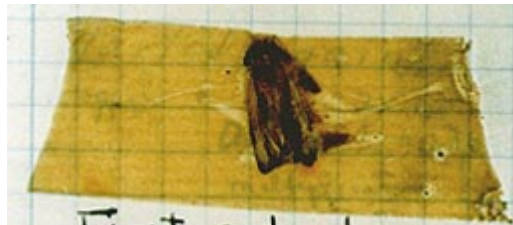
Servir

La répétition

- Il faut faire attention à ce que la condition puisse devenir vraie
 - Boucle infinie !

```
TANT QUE tu ne touches pas le mur  
  fais un pas en avant  
  fais un pas en arrière
```

- C'est un « bug »



Le branchement

- Instruction qui brise le déroulement séquentiel d'un algorithme
 - En pseudo code, pour désigner un emplacement, on utilise le numéro de ligne

ALLER EN ...

- Le branchement conditionnel associe une condition à un branchement

SI Condition ALORS ALLER EN ...

- *Son utilisation n'est pas recommandée (par moi)*
 - On peut souvent s'en passer
 - Son utilisation est un défaut de conception de l'algorithme
 - Seuls cas pratiques utiles : `break` et `continue`

Le branchement

- Branchements usuels : `break` et `continue`

- `break` : interruption définitive d'une boucle
- `continue` : saut immédiat à la fin du tour de boucle

REPETER

A

SI Urgence **ALORS**

break

B

JUSQU'À CE QUE condition

suite

Exemple d'exécution :

A

B

A

B

A

B

A

suite

Urgence !

Le branchement

- Branchements usuels : `break` et `continue`

- `break` : interruption définitive d'une boucle
- `continue` : saut immédiat à la fin du tour de boucle

REPETER

A

SI Urgence **ALORS**

`continue`

B

JUSQU'À CE QUE condition

suite

Exemple d'exécution :

A

B

A

B

A

B

A

A

B

A

B

(...)

Urgence !

Bilan

- Quatre notions d'algorithmique
 - les instructions
 - les structures de contrôles : séquence, alternative, répétition et branchement
 - les conditions
 - les procédures (pas vu ici!)
- Programmer
 - Donner une description abstraite de la séquence d'actions élémentaires (les instructions) que doit réaliser un robot
- En programmation impérative, l'algorithmique consiste à
 - décomposer une tâche en sous-tâches jusqu'aux actions élémentaires
 - La décomposition se fait *selon* les structures de contrôles