

Introduction à l'informatique en BCPST

Alexandre Benoit

BCPST

« L'enseignement de l'informatique en classes préparatoires de la filière BCPST a pour objectif d'introduire puis de consolider les concepts de base de l'informatique, à savoir l'analyse et la conception de processus de raisonnement automatisé, c'est-à-dire des algorithmes, et la question de la représentation des données. »

Programme officiel 2013.

« L'enseignement de l'informatique en classes préparatoires de la filière BCPST a pour objectif d'introduire puis de consolider les concepts de base de l'informatique, à savoir l'analyse et la conception de processus de raisonnement automatisé, c'est-à-dire des algorithmes, et la question de la représentation des données. »

Programme officiel 2013.

En pratique :

- On étudiera et maîtrisera certains algorithmes fondamentaux

« L'enseignement de l'informatique en classes préparatoires de la filière BCPST a pour objectif d'introduire puis de consolider les concepts de base de l'informatique, à savoir l'analyse et la conception de processus de raisonnement automatisé, c'est-à-dire des algorithmes, et la question de la représentation des données. »

Programme officiel 2013.

En pratique :

- On étudiera et maîtrisera certains algorithmes fondamentaux
- On apprendra à programmer dans le langage **Python**.

« L'enseignement de l'informatique en classes préparatoires de la filière BCPST a pour objectif d'introduire puis de consolider les concepts de base de l'informatique, à savoir l'analyse et la conception de processus de raisonnement automatisé, c'est-à-dire des algorithmes, et la question de la représentation des données. »

Programme officiel 2013.

En pratique :

- On étudiera et maîtrisera certains algorithmes fondamentaux
- On apprendra à programmer dans le langage **Python**.
- On appliquera le tout à des problèmes concrets.

- Épreuve **obligatoire** à l'oral du concours, en même temps que l'épreuve de mathématiques,

- Épreuve **obligatoire** à l'oral du concours, en même temps que l'épreuve de mathématiques,
- En première année : cours d'introduction,

- Épreuve **obligatoire** à l'oral du concours, en même temps que l'épreuve de mathématiques,
- En première année : cours d'introduction,
- En deuxième année : Réalisation d'un projet,

- Épreuve **obligatoire** à l'oral du concours, en même temps que l'épreuve de mathématiques,
- En première année : cours d'introduction,
- En deuxième année : Réalisation d'un projet,
- Au concours : exercice et exposée de la solution du projet avec examen du code imprimé sur papier.

- Épreuve **obligatoire** à l'oral du concours, en même temps que l'épreuve de mathématiques,
- En première année : cours d'introduction,
- En deuxième année : Réalisation d'un projet,
- Au concours : exercice et exposée de la solution du projet avec examen du code imprimé sur papier.

- Épreuve **obligatoire** à l'oral du concours, en même temps que l'épreuve de mathématiques,
- En première année : cours d'introduction,
- En deuxième année : Réalisation d'un projet,
- Au concours : exercice et exposée de la solution du projet avec examen du code imprimé sur papier.

Comparable au TIPE

Qu'est ce qu'un ordinateur ?

« Souvent, quelques-unes des caractéristiques fondamentales nécessaires pour être considérées comme un ordinateur sont :

- 1 qu'il soit électronique,

Qu'est ce qu'un ordinateur ?

« Souvent, quelques-unes des caractéristiques fondamentales nécessaires pour être considérées comme un ordinateur sont :

- ① qu'il soit électronique,
- ② numérique (au lieu d'analogique),

Qu'est ce qu'un ordinateur ?

« Souvent, quelques-unes des caractéristiques fondamentales nécessaires pour être considérées comme un ordinateur sont :

- ① qu'il soit électronique,
- ② numérique (au lieu d'analogique),
- ③ qu'il soit programmable,

Qu'est ce qu'un ordinateur ?

« Souvent, quelques-unes des caractéristiques fondamentales nécessaires pour être considérées comme un ordinateur sont :

- ① qu'il soit électronique,
- ② numérique (au lieu d'analogique),
- ③ qu'il soit programmable,
- ④ qu'il puisse exécuter les quatre opérations élémentaires (addition, soustraction, multiplication, division) et -souvent- qu'il puisse extraire une racine carrée ou adresser une table qui en contient,

Qu'est ce qu'un ordinateur ?

« Souvent, quelques-unes des caractéristiques fondamentales nécessaires pour être considérées comme un ordinateur sont :

- ① qu'il soit électronique,
- ② numérique (au lieu d'analogique),
- ③ qu'il soit programmable,
- ④ qu'il puisse exécuter les quatre opérations élémentaires (addition, soustraction, multiplication, division) et -souvent- qu'il puisse extraire une racine carrée ou adresser une table qui en contient,
- ⑤ qu'il puisse exécuter des programmes enregistrés en mémoire. »

Qu'est ce qu'un ordinateur ?

« Souvent, quelques-unes des caractéristiques fondamentales nécessaires pour être considérées comme un ordinateur sont :

- 1 qu'il soit électronique,
- 2 numérique (au lieu d'analogique),
- 3 qu'il soit programmable,
- 4 qu'il puisse exécuter les quatre opérations élémentaires (addition, soustraction, multiplication, division) et -souvent- qu'il puisse extraire une racine carrée ou adresser une table qui en contient,
- 5 qu'il puisse exécuter des programmes enregistrés en mémoire. »

Qu'est ce qu'un ordinateur ?

« Souvent, quelques-unes des caractéristiques fondamentales nécessaires pour être considérées comme un ordinateur sont :

- ① qu'il soit électronique,
- ② numérique (au lieu d'analogique),
- ③ qu'il soit programmable,
- ④ qu'il puisse exécuter les quatre opérations élémentaires (addition, soustraction, multiplication, division) et -souvent- qu'il puisse extraire une racine carrée ou adresser une table qui en contient,
- ⑤ qu'il puisse exécuter des programmes enregistrés en mémoire. »

Bernard Cohen (2000)

Qu'est ce qu'un ordinateur ?

« Souvent, quelques-unes des caractéristiques fondamentales nécessaires pour être considérées comme un ordinateur sont :

- 1 qu'il soit électronique,
- 2 numérique (au lieu d'analogique),
- 3 qu'il soit programmable,
- 4 qu'il puisse exécuter les quatre opérations élémentaires (addition, soustraction, multiplication, division) et -souvent- qu'il puisse extraire une racine carrée ou adresser une table qui en contient,
- 5 qu'il puisse exécuter des programmes enregistrés en mémoire. »

Bernard Cohen (2000)

Exemple

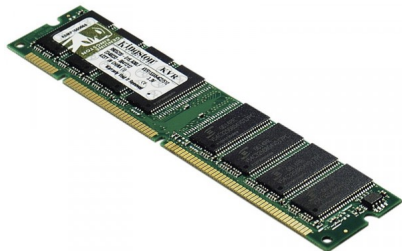
Les ordinateurs de bureau, les ordinateurs portables mais aussi les tablettes ou les smartphones sont des ordinateurs.

Les composants essentiels de l'ordinateur sont :

- La mémoire vive ;
- Le processeur ;
- La mémoire de masse ;
- Carte graphique ;
- Périphériques.

Les composants essentiels de l'ordinateur sont :

- **La mémoire vive ;**
 - Pas de sens à priori ;
 - Inertie ;
 - Accès direct.
- Le processeur ;
- La mémoire de masse ;
- Carte graphique ;
- Périphériques.



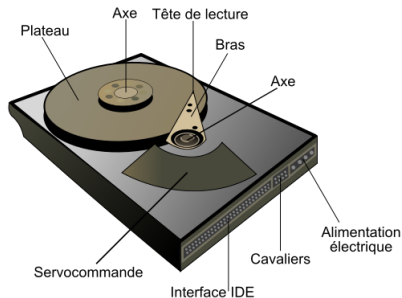
Les composants essentiels de l'ordinateur sont :

- La mémoire vive ;
- **Le processeur** ;
 - Arithmétique de base (+, -, ×, ÷) et opérations logiques (disjonction, conjonction, négation) sur des valeurs de vérité ;
 - Toute petite mémoire mais avec un accès très rapide (plus rapide que la mémoire vive) ;
 - Accès direct à la mémoire vive.
- La mémoire de masse ;
- Carte graphique ;
- Périphériques.



Les composants essentiels de l'ordinateur sont :

- La mémoire vive ;
- Le processeur ;
- **La mémoire de masse ;**
 - Mémoire non volatile (elle reste en place après une coupure de courant) ;
 - Grande capacité ;
 - Accès très lent (1000 fois plus lent que la mémoire vive) ;
 - Différents aspects : Disque dur (ordinateur), mémoire flash (smartphone), CD Rom, Clé USB, etc ...
- Carte graphique ;
- Périphériques.



Les composants essentiels de l'ordinateur sont :

- La mémoire vive ;
- Le processeur ;
- La mémoire de masse ;
- **Carte graphique** ;
Tous les calculs liés à l'affichage sont délégués à la carte graphique.
- Périphériques.



Les composants essentiels de l'ordinateur sont :

- La mémoire vive ;
- Le processeur ;
- La mémoire de masse ;
- Carte graphique ;
- **Périphériques.**
Clavier, Lecteur CD, Souris,
Moniteur, Enceinte, Scanner,
Webcam, Micro etc ..

Architecture : Partie matérielle.

Système d'exploitation

Architecture : Partie matérielle.

Système d'exploitation : Partie logicielle.

Système d'exploitation

Architecture : Partie matérielle.

Système d'exploitation : Partie logicielle. Il existe deux grandes familles de système d'exploitation

Exemple

- Les systèmes issues d'**UNIX** (Mac OS, IOS, GNU/Linux, Android)
- Les systèmes issues de la famille **Microsoft Windows**

Système d'exploitation

Architecture : Partie matérielle.

Système d'exploitation : Partie logicielle. Il existe deux grandes familles de système d'exploitation

Exemple

- Les systèmes issues d'UNIX (Mac OS, IOS, GNU/Linux, Android)
- Les systèmes issues de la famille Microsoft Windows

Le système d'exploitation a les **responsabilités suivantes** :

- Donner l'illusion que l'ordinateur est multitâche ;

Architecture : Partie matérielle.

Système d'exploitation : Partie logicielle. Il existe deux grandes familles de système d'exploitation

Exemple

- Les systèmes issues d'UNIX (Mac OS, IOS, GNU/Linux, Android)
- Les systèmes issues de la famille Microsoft Windows

Le système d'exploitation a les responsabilités suivantes :

- Donner l'illusion que l'ordinateur est multitâche ;
- Identifier les utilisateurs ;

Architecture : Partie matérielle.

Système d'exploitation : Partie logicielle. Il existe deux grandes familles de système d'exploitation

Exemple

- Les systèmes issues d'UNIX (Mac OS, IOS, GNU/Linux, Android)
- Les systèmes issues de la famille Microsoft Windows

Le système d'exploitation a les responsabilités suivantes :

- Donner l'illusion que l'ordinateur est multitâche ;
- Identifier les utilisateurs ;
- Gérer l'organisation du disque dur et de ses fichiers ;

Architecture : Partie matérielle.

Système d'exploitation : Partie logicielle. Il existe deux grandes familles de système d'exploitation

Exemple

- Les systèmes issues d'UNIX (Mac OS, IOS, GNU/Linux, Android)
- Les systèmes issues de la famille Microsoft Windows

Le système d'exploitation a les responsabilités suivantes :

- Donner l'illusion que l'ordinateur est multitâche ;
- Identifier les utilisateurs ;
- Gérer l'organisation du disque dur et de ses fichiers ;
- Contrôler l'accès aux données du disque ;

Architecture : Partie matérielle.

Système d'exploitation : Partie logicielle. Il existe deux grandes familles de système d'exploitation

Exemple

- Les systèmes issues d'UNIX (Mac OS, IOS, GNU/Linux, Android)
- Les systèmes issues de la famille Microsoft Windows

Le système d'exploitation a les responsabilités suivantes :

- Donner l'illusion que l'ordinateur est multitâche ;
- Identifier les utilisateurs ;
- Gérer l'organisation du disque dur et de ses fichiers ;
- Contrôler l'accès aux données du disque ;
- Gérer le lancement des différentes applications utilisées ;

Système d'exploitation

Architecture : Partie matérielle.

Système d'exploitation : Partie logicielle. Il existe deux grandes familles de système d'exploitation

Exemple

- Les systèmes issues d'UNIX (Mac OS, IOS, GNU/Linux, Android)
- Les systèmes issues de la famille Microsoft Windows

Le système d'exploitation a les responsabilités suivantes :

- Donner l'illusion que l'ordinateur est multitâche ;
- Identifier les utilisateurs ;
- Gérer l'organisation du disque dur et de ses fichiers ;
- Contrôler l'accès aux données du disque ;
- Gérer le lancement des différentes applications utilisées ;
- Servir de garde-fou en cas de tentative de mauvaise utilisation des ressources de l'ordinateur

- Sous Linux, tous les fichiers sont regroupés dans une unique arborescence. Le sommet de l'arbre est un répertoire appelé / . Cette racine possède plusieurs sous-répertoire, dont généralement un appelé home.

- Sous Linux, tous les fichiers sont regroupés dans une unique arborescence. Le sommet de l'arbre est un répertoire appelé / . Cette racine possède plusieurs sous-répertoire, dont généralement un appelé home.
- Pour se repérer, on utilise le chemin **absolu** du fichier (ex : /home/login/travail/td1/data/img.jpeg) ou le chemin **relatif** (ex :data/img.jpeg)

Dans un système GNU/Linux, tout fichier se voit attribuer des droits pour 3 identités :

- le propriétaire - c'est l'utilisateur qui a créé le fichier ou l'utilisateur que root a désigné comme propriétaire
- le groupe (qui n'est pas forcément le groupe du propriétaire)
- les autres (ceux qui ne font pas partie du groupe)

Dans un système GNU/Linux, tout fichier se voit attribuer des droits pour 3 identités :

- le propriétaire - c'est l'utilisateur qui a créé le fichier ou l'utilisateur que root a désigné comme propriétaire
- le groupe (qui n'est pas forcément le groupe du propriétaire)
- les autres (ceux qui ne font pas partie du groupe)

La commande `ls -l` nous permet d'afficher les droits d'un fichier sous GNU/Linux.

Pour chaque identité (voir plus haut), il existe 3 droits d'accès :

- r - read (le droit de lecture)
- w - write (le droit d'écriture)
- x - execute (le droit d'exécution)

- Python est un langage de programmation moderne, utilisé de plus en plus dans **l'industrie** et dans **l'éducation**,

- Python est un langage de programmation moderne, utilisé de plus en plus dans l'industrie et dans l'éducation,
- Ce langage est :

- Python est un langage de programmation moderne, utilisé de plus en plus dans l'industrie et dans l'éducation,
- Ce langage est :
 - **Multi-plateforme** : Linux, Windows, Mac OS, Android, etc.

- Python est un langage de programmation moderne, utilisé de plus en plus dans l'industrie et dans l'éducation,
- Ce langage est :
 - Multi-plateforme : Linux, Windows, Mac OS, Android, etc.
 - **Libre** : Toute utilisation (y compris commerciale) possible, accès au code source, maintenue par une fondation.

- Python est un langage de programmation moderne, utilisé de plus en plus dans l'industrie et dans l'éducation,
- Ce langage est :
 - Multi-plateforme : Linux, Windows, Mac OS, Android, etc.
 - Libre : Toute utilisation (y compris commerciale) possible, accès au code source, maintenue par une fondation.
 - **interprété** instructions exécutées une par une, accès direct par une ligne de commande

- Python est un langage de programmation moderne, utilisé de plus en plus dans l'industrie et dans l'éducation,
- Ce langage est :
 - Multi-plateforme : Linux, Windows, Mac OS, Android, etc.
 - Libre : Toute utilisation (y compris commerciale) possible, accès au code source, maintenue par une fondation.
 - interprété instructions exécutées une par une, accès direct par une ligne de commande
 - **haut niveau** Proche du langage humain

- Python est un langage de programmation moderne, utilisé de plus en plus dans l'industrie et dans l'éducation,
- Ce langage est :
 - Multi-plateforme : Linux, Windows, Mac OS, Android, etc.
 - Libre : Toute utilisation (y compris commerciale) possible, accès au code source, maintenue par une fondation.
 - interprété instructions exécutées une par une, accès direct par une ligne de commande
 - haut niveau Proche du langage humain
 - **Modulaire** Bibliothèque pour le calcul scientifique, le graphique, l'analyse de signaux et d'images, etc.

- Python est un langage de programmation moderne, utilisé de plus en plus dans l'industrie et dans l'éducation,
- Ce langage est :
 - Multi-plateforme : Linux, Windows, Mac OS, Android, etc.
 - Libre : Toute utilisation (y compris commerciale) possible, accès au code source, maintenue par une fondation.
 - interprété instructions exécutées une par une, accès direct par une ligne de commande
 - haut niveau Proche du langage humain
 - Modulaire Bibliothèque pour le calcul scientifique, le graphique, l'analyse de signaux et d'images, etc.
 - **Très simple** à prendre en main (tout se fait sur l'indentation du programme).

La fenêtre de Pyzo est divisée en trois parties

- **L'éditeur** à gauche, dans lequel on écrira les programmes.
- **La console interactive** en haut à droite, dans laquelle s'exécuteront les programmes.
- **Le gestionnaire de fichiers** qui nous servira à naviguer entre les répertoires.

Taper dans la console

2+2

La console interactive

Taper dans la console

2+2

a=2

puis

a+a

Taper dans la console

2+2

a=2

puis

a+a

Taper ensuite

b+1

On obtient alors une erreur, la dernière ligne du message indique d'où vient l'erreur.

La console interactive

Taper dans la console

2+2

a=2

puis

a+a

Taper ensuite

b+1

On obtient alors une erreur, la dernière ligne du message indique d'où vient l'erreur.

Quels sont les défauts de cette console ?

Avec l'éditeur, on peut **sauvegarder** son programme et le **modifier facilement**

Avec l'éditeur, on peut sauvegarder son programme et le modifier facilement

Essayer :

```
print (" Bonjour ")
```

```
x=42
```

```
print (x)
```

Avec l'éditeur, on peut sauvegarder son programme et le modifier facilement

Essayer :

```
print (" Bonjour ")
```

```
x=42
```

```
print (x)
```

La première chose à faire quand on commence un programme est de le **sauvegarder**.

Avec l'éditeur, on peut sauvegarder son programme et le modifier facilement

Essayer :

```
print (" Bonjour ")
```

```
x=42
```

```
print (x)
```

La première chose à faire quand on commence un programme est de le sauvegarder.

Pour exécuter ce travail, on utilise la commande **Ctrl-Shift-E** ou l'icône avec le *petit bonhomme qui court*.

On prend l'exemple suivant :

$$x=10$$

$$y=7$$

$$x=x+y$$

$$y=x$$

$$x=5/(x-y)$$

On remarque que cet exemple provoque une erreur.

On prend l'exemple suivant :

$$x=10$$

$$y=7$$

$$x=x+y$$

$$y=x$$

$$x=5/(x-y)$$

On remarque que cet exemple provoque une erreur.

On prend l'exemple suivant :

$x=10$

$y=7$

$x=x+y$

$y=x$

$x=5/(x-y)$

On remarque que cet exemple provoque une erreur.

Dans Pyzo, il existe un débogueur qui permet de savoir d'où vient l'erreur.

- *Ajouter un point d'arrêt* en cliquant dans le marge gauche du programme à côté de la ligne que l'on soupçonne poser problème.
- Lancer le programme avec Ctrl-Shift-E

On prend l'exemple suivant :

$x=10$

$y=7$

$x=x+y$

$y=x$

$x=5/(x-y)$

On remarque que cet exemple provoque une erreur.

Dans Pyzo, il existe un débogueur qui permet de savoir d'où vient l'erreur.

- *Ajouter un point d'arrêt* en cliquant dans le marge gauche du programme à côté de la ligne que l'on soupçonne poser problème.
- Lancer le programme avec Ctrl-Shift-E
- Choisir la fleche faisant un demi-cercle dans le shells pour aller ligne à ligne à partir du point d'arrêt.

- 1 Taper le programme suivant dans l'éditeur et l'exécuter. Que se passe-t'il ?

```
i=10
while i!=0:
    i = 1-i
print("Ce programme est fini")
```

Un premier exercice

- 1 Taper le programme suivant dans l'éditeur et l'exécuter. Que se passe-t'il ?

```
i=10
while i!=0:
    i = 1-i
print ("Ce programme est fini")
```

- 2 Laisser un point d'arrêt à la première ligne.

Un premier exercice

- 1 Taper le programme suivant dans l'éditeur et l'exécuter. Que se passe-t'il ?

```
i=10
while i!=0:
    i = 1-i
print ("Ce programme est fini ")
```

- 2 Laisser un point d'arrêt à la première ligne.
- 3 Exécuter ce programme pas à pas et observer les valeurs successives prises par la variable i en allant dans Window puis Workspace. Expliquer le comportement observé à la première question.

Un premier exercice

- 1 Taper le programme suivant dans l'éditeur et l'exécuter. Que se passe-t'il ?

```
i=10
while i!=0:
    i = 1-i
print ("Ce programme est fini ")
```

- 2 Laisser un point d'arrêt à la première ligne.
- 3 Exécuter ce programme pas à pas et observer les valeurs successives prises par la variable i en allant dans Window puis Workspace. Expliquer le comportement observé à la première question.
- 4 Proposer une correction à ce programme pour qu'il termine.

Quelques références :

- Sur mon site vous trouverez les informations pour installer cet éditeur : <http://alexandre.benoit.83.free.fr/BCPST/>
- L'informatique pour tous de Wack et al aux éditions Eyrolles (disponible gratuitement sur le web)
https://wiki.inria.fr/sciencinfolycee/Fichier:Informatique_pour_tous_en_classes_pr%C3%A9paratoires_aux_grandes_%C3%A9coles.pdf. (Ce cours est largement inspiré de ce livre).
Attention : une grosse partie de ce livre n'est pas au programme BCPST
- Le site [openclassrooms](#) propose un bon tutoriel sur Python (et d'autres) et démarre d'un niveau basique.