

# Présentation et Révisions

# Introduction : l'informatique en BCPST 2

présente dans :

- 1 l'épreuve de modélisation,

# Introduction : l'informatique en BCPST 2

présente dans :

- ① l'épreuve de modélisation,
- ② l'oral de mathématique,

# Introduction : l'informatique en BCPST 2

présente dans :

- ① l'épreuve de modélisation,
- ② l'oral de mathématique,
- ③ un projet travaillé pendant l'année.

# Organisation de l'année

- ➊ Jusqu'à la Toussaint : révisions et compléments d'algorithmique et de programmation.
- ➋ Toussaint-Noël : exercices d'informatique et travail sur projet.
- ➌ Janvier-fin : exercices d'informatique et travail sur projet.

- 1 Révisions : programmation de base

# Jusqu'à la Toussaint

- ① Révisions : programmation de base
- ② Révisions : les listes, analyse numérique, probabilités

# Jusqu'à la Toussaint

- ① Révisions : programmation de base
- ② Révisions : les listes, analyse numérique, probabilités
- ③ Compléments sur les listes



# Jusqu'à la Toussaint

- ① Révisions : programmation de base
- ② Révisions : les listes, analyse numérique, probabilités
- ③ Compléments sur les listes
- ④ La récursivité

# Jusqu'à la Toussaint

- ① Révisions : programmation de base
- ② Révisions : les listes, analyse numérique, probabilités
- ③ Compléments sur les listes
- ④ La récursivité
- ⑤ Les algorithmes de tris

# Jusqu'à la Toussaint

- ① Révisions : programmation de base
- ② Révisions : les listes, analyse numérique, probabilités
- ③ Compléments sur les listes
- ④ La récursivité
- ⑤ Les algorithmes de tris
- ⑥ Une nouvelle structure de données : les graphes

# Jusqu'à la Toussaint

- 1 Révisions : programmation de base
- 2 Révisions : les listes, analyse numérique, probabilités
- 3 Compléments sur les listes
- 4 La récursivité
- 5 Les algorithmes de tris
- 6 Une nouvelle structure de données : les graphes
- 7 L'algorithme de Dijkstra

- 1 Révisions : programmation de base
- 2 Révisions : les listes, analyse numérique, probabilités
- 3 Compléments sur les listes
- 4 La récursivité
- 5 Les algorithmes de tris
- 6 Une nouvelle structure de données : les graphes
- 7 L'algorithme de Dijkstra
- 8 Projet : s'organiser.

- Constitution des groupes, choix du projet (par 3).

- Constitution des groupes, choix du projet (par 3).
- Idée précise du projet dès les vacances de la Toussaint.

- Constitution des groupes, choix du projet (par 3).
- Idée précise du projet dès les vacances de la Toussaint.
- Début des projets dès le retour des vacances.



Rappels (mots clés) :

- ① instructions conditionnelles `if` et `else`,
- ② boucles `while` et `for`,
- ③ fonctions,
- ④ variables locales et globales.

# Instruction conditionnelle

Syntaxe :

```
if condition 1 :  
    instruction 1  
elif condition 2 :  
    instruction 2  
.  
.  
.  
else :  
    instruction n
```

Exemple :

```
if x%2==0 :  
    print("x est pair")  
else :  
    print("x est impair")
```

# Boucle while

Syntaxe :

```
while condition :  
    instruction
```

Exemple :

```
n=1  
p=1  
while n<1000000 :  
    p=p+1  
    n=n*p  
print(p)
```

# Boucle for

- Syntaxe :

```
for i in range(debut,fin,pas) :  
    instruction
```

# Boucle for

- Syntaxe :

```
for i in range(debut,fin,pas) :  
    instruction
```

- Exemple :

```
for i in range(9,-18,-6) :  
    print(i)
```

- Syntaxe :

```
def fonction(a1,a2,...,an):  
    instruction 1  
    instruction 2  
    instruction 3  
    return
```

- Syntaxe :

```
def fonction(a1,a2,...,an):  
    instruction 1  
    instruction 2  
    instruction 3  
    return
```

- Exemple :

```
def mystere(n) :  
    S=0  
    for i in range(0,n) :  
        S=S+i*i  
    return S
```

# Variable globale, variable locale

Différences entre :

```
a=1
def hello() :
    a='bonjour'
    return
print(a)
```

```
a=1
def hello() :
    global a
    a='bonjour'
    return
print(a)
```