

Stage pratique de 3 jour(s)  
Réf : INP

### Participants

Toute personne devant apprendre à programmer.

### Pré-requis

Aucune connaissance particulière.

Prix 2012 : 1650€ HT

### Dates des sessions

#### Paris

5 mar. 2012, 2 avr. 2012  
2 mai 2012, 4 juin 2012  
2 juil. 2012, 20 août. 2012  
3 sep. 2012, 1 oct. 2012  
5 nov. 2012, 3 déc. 2012

# Introduction à la programmation

## OBJECTIFS

Ce stage vous permettra de comprendre les fondements de la programmation et de l'algorithmique. Vous acquérez des bases en programmation qui vous permettront d'aborder n'importe quel langage dans les meilleures conditions. Tous les aspects essentiels seront vus : les modèles de programmation, les éléments de lexique et de syntaxe, les outils, l'organisation du code, l'accès aux bases de données et les tests.

### 1) Un programme

### 2) Nécessité d'un algorithme

### 3) Genèse d'un premier programme

### 4) Règles de programmation

### 5) Les variables

### 6) Opérateurs et expressions

### 7) Les structures de contrôle

### 8) Les procédures et les fonctions

### 9) Introduction à la programmation objet

### 10) L'accès aux bases de données

### 11) Maintenance, débogage et test des programmes

## Travaux pratiques

Ce stage contient plus de 60 % de travaux pratiques effectués en Visual Basic.

## 1) Un programme

- Qu'est-ce qu'un programme ?
- Qu'est-ce qu'un langage ? Les différents paradigmes.
- Quel langage pour quelle application ?
- Les compilateurs. Les exécutables.
- Les responsabilités d'un programmeur.

### Travaux pratiques

Présentation de différents langages (Java, Visual Basic, C, C++).

## 2) Nécessité d'un algorithme

- Qu'est-ce qu'un algorithme ?
- Les besoins auxquels répond un algorithme.
- Le concept de pseudo-langage.

### Travaux pratiques

Écriture d'un premier algorithme en pseudo-langage.

## 3) Genèse d'un premier programme

- Écriture d'un programme simple.
- Compilation et exécution du programme.
- Qu'est-ce qu'une librairie ? Son rôle, son usage.

### Travaux pratiques

Découverte de l'environnement de développement et d'exécution. Écriture, compilation et exécution d'un premier programme.

## 4) Règles de programmation

- Convention de nommage.
- Convention syntaxique.
- Utilisation des commentaires. Pourquoi commenter les développements ?
- Améliorer la lisibilité des programmes : indentation du code, découpage du code...

## 5) Les variables

- Qu'est-ce qu'une variable ?
- Pourquoi typer une variable ?
- Les types primitifs : entiers, chaînes de caractères, nombres réels, autres.
- Déclaration, définition et initialisation d'une variable.
- Les constantes.
- Saisie, affichage, affectation, conversion de type.
- Organiser ses données sous forme de tableaux.
- Les types évolués : enregistrement, matrice, arbre.

### Travaux pratiques

Écriture de plusieurs programmes simples manipulant les variables.

## 6) Opérateurs et expressions

- Les différents opérateurs (multiplicatif, additif, comparaison, égalité, logique, affectation).
- Combinaison d'opérateurs.
- Expression booléenne.

**Travaux pratiques**

*Manipulation des opérateurs et des expressions booléennes.*

## 7) Les structures de contrôle

- Les sélections alternatives (si, si-alors-sinon, sélection cas).
- Les séquences (notion de Début... Fin).
- Les boucles itératives (tant que-répéter, répéter-jusqu'à, pour-de-à).
- Imbrication des instructions.
- Les commentaires.

**Travaux pratiques**

*Utilisation des structures de contrôle pour implémenter un algorithme.*

## 8) Les procédures et les fonctions

- Définitions : procédure, fonction.
- Pourquoi sont-elles incontournables en programmation (réutilisabilité, lisibilité...) ?
- Le passage de paramètres.
- Le code retour d'une fonction.
- Sensibilisation aux limites du passage de la valeur d'une variable.
- Notion de passage par adresse.
- Appel de fonctions.

**Travaux pratiques**

*Debugging de programmes exemples.*

## 9) Introduction à la programmation objet

- Les concepts associés à la programmation objet : classe, attribut, méthode, argument.
- La modélisation objet à partir des exigences fonctionnelles : introduction aux bonnes pratiques d'organisation de conception et d'organisation d'un programme.

**Travaux pratiques**

*Illustration des concepts objets.*

## 10) L'accès aux bases de données

- Organisation et stockage des données.
- Les traitements de base (connexion, requêtes, récupération des données).
- Application cliente et serveur de données.
- Affichage et manipulation des données dans l'application cliente.

**Travaux pratiques**

*Création d'un formulaire de recherche d'informations dans une base de données.*

## 11) Maintenance, débogage et test des programmes

- Savoir lire et interpréter les différents messages d'erreurs.
- Utiliser un débogueur : exécution d'un programme pas à pas, points d'arrêts, inspecter les variables pendant l'exécution.
- Prévoir les tests unitaires.

**Travaux pratiques**

*Utilisation d'un débogueur pour contrôler l'exécution des programmes.*