Fichelo

Licence 1 · Informatique

Informatique L1 : fiches de révision

Le tronc commun de première année : représentation de l'information, architecture, logique, Python, algorithmique, structures de données

28	114	6
fiches de révision	exercices corrigés	chapitres

D'après : Fiche nationale RNCP40116 — Licence Informatique (France Compétences) : compétences spécifiques de la mention ; Arrêté du 30 juillet 2018 relatif au diplôme national de licence (180 crédits ECTS, référentiels de compétences des mentions) ; Documentation officielle de Python 3 (types natifs, opérateurs, exceptions). Année 2026-2027.

fichelo.fr · usage personnel et familial

Sommaire

REPRÉSENTATION DE L'INFORMATION

1 Bases 2, 10 et 16 : écrire et convertir
2 Entiers signés : le complément à deux
3 Nombres à virgule flottante (aperçu IEEE 754)
4 Caractères : ASCII, Unicode, UTF-8

ARCHITECTURE DE L'ORDINATEUR

5 Le modèle de von Neumann
6 Un petit langage machine

LOGIQUE BOOLÉENNE ET CIRCUITS

7 Algèbre de Boole et tables de vérité
8 Formes normales disjonctive et conjonctive
9 Portes logiques, demi-additionneur et additionneur

PROGRAMMER EN PYTHON

10 Types, variables et expressions
11 Conditions et boucles
12 Fonctions et portée des variables
13 Chaînes de caractères
14 Listes, tuples, mutabilité et alias
15 Dictionnaires et ensembles
16 Listes en compréhension et fonctions d'ordre supérieur
17 Exceptions, assertions et tests
18 Fichiers et ligne de commande

ALGORITHMIQUE

19 Prouver une boucle : invariant et terminaison
20 Complexité : compter les opérations
21 Recherche séquentielle et recherche dichotomique
22 Tris par sélection et par insertion
23 Récursivité
24 Diviser pour régner : le tri fusion

STRUCTURES DE DONNÉES

25 Piles et files
26 Listes chaînées
27 Tables de hachage (les dictionnaires vus de l'intérieur)
28 Introduction aux arbres binaires

Comment utiliser ce pack. Il n'existe pas de programme national de licence : chaque université construit ses unités d'enseignement à partir des compétences de la mention Informatique. Ces fiches couvrent le socle que l'on retrouve en première année presque partout ; **l'ordre, l'intitulé des UE et le langage utilisé** (Python le plus souvent, parfois C ou OCaml) **varient selon votre université** : suivez le plan de votre cours et piochez la fiche correspondante. Une fiche = une notion : lisez « À retenir », refaites l'exemple sans regarder, puis les exercices ; les corrigés sont à la fin. Tout le code Python du pack a été exécuté et testé (Python 3.12).

FICHE 1 · REPRÉSENTATION DE L'INFORMATION

Bases 2, 10 et 16 : écrire et convertir

Au programme : Représenter les entiers naturels en binaire et en hexadécimal, passer d'une base à l'autre (compétence « éléments fonctionnels d'un ordinateur » de la mention).

À RETENIR

- En base *b*, le nombre écrit $a_{n-1}…a_1a_0$ vaut $a_{n-1}\cdot b^{n-1} + … + a_1\cdot b + a_0$, avec des chiffres $0 \le a_i < b$.
- **Base 2 → base 10** : on additionne les puissances de 2 des bits à 1. $1011_2 = 8 + 2 + 1 = 11$.
- **Base 10 → base 2** : divisions successives par 2, on lit les restes **de bas en haut** (du dernier au premier).
- **Hexadécimal** : chiffres 0-9 puis A=10 … F=15. Un chiffre hexadécimal = exactement 4 bits : on convertit binaire ↔ hexa par paquets de 4 bits en partant de la droite.
- Avec *n* bits on code 2^n valeurs, de 0 à $2^n - 1$ (8 bits : 0 à 255 ; 16 bits : 0 à 65 535).
- Puissances à connaître : $2^8 = 256$, $2^{10} = 1\,024$, $2^{16} = 65\,536$, $2^{32} \approx 4{,}3$ milliards.
- En Python : `bin(11)` → `'0b1011'`, `hex(255)` → `'0xff'`, `int('1011', 2)` → 11, littéraux `0b1011` et `0xFF`.

VOCABULAIRE

bit	chiffre binaire (0 ou 1)
octet	groupe de 8 bits
bit de poids fort	bit le plus à gauche (plus grande puissance de 2)

EXEMPLE CORRIGÉ

Convertir 156 en binaire puis en hexadécimal.

156 = 2×78 + 0 ; 78 = 2×39 + 0 ; 39 = 2×19 + 1 ; 19 = 2×9 + 1 ; 9 = 2×4 + 1 ; 4 = 2×2 + 0 ; 2 = 2×1 + 0 ; 1 = 2×0 + 1.

Restes lus de bas en haut : **10011100_2**. Vérification : 128 + 16 + 8 + 4 = 156. Paquets de 4 : 1001 | 1100 = 9 | C, donc **0x9C**.

ATTENTION

- Lire les restes dans l'ordre d'apparition donne le nombre à l'envers.
- Grouper les bits par 4 en partant de la **gauche** : faux dès que le nombre de bits n'est pas multiple de 4 (101 0110 = 0x56, pas 0xA6).
- `bin()` et `hex()` renvoient des **chaînes**, pas des nombres.

JE M'ENTRAÎNE

1. Convertir $1101\ 0110_2$ en décimal et en hexadécimal.
2. Convertir 2026 en binaire.
3. Convertir 0x3F2 en binaire puis en décimal.
4. Combien de bits faut-il au minimum pour coder les entiers de 0 à 1000 ?
5. Que valent en Python `int('ff', 16) + 0b101` et `hex(4096)` ?

fichelo.fr · usage personnel et familial

Sources

Les fiches suivent les textes officiels ci-dessous (programme en vigueur pour l'année indiquée sur la couverture). Fichelo est un site indépendant : il n'est affilié ni au ministère de l'Éducation nationale, ni au ministère de l'Enseignement supérieur, ni à aucun éditeur.

1. Fiche nationale RNCP40116 — Licence Informatique (France Compétences) : compétences spécifiques de la mention
   https://www.francecompetences.fr/recherche/rncp/40116/
2. Arrêté du 30 juillet 2018 relatif au diplôme national de licence (180 crédits ECTS, référentiels de compétences des mentions)
   https://www.legifrance.gouv.fr/loda/id/JORFTEXT000037291166
3. Documentation officielle de Python 3 (types natifs, opérateurs, exceptions)
   https://docs.python.org/fr/3/library/stdtypes.html

Ce PDF est vendu pour un usage personnel et familial (ou pour la classe de l'enseignant acheteur). Merci de ne pas le diffuser en ligne. Version du 2026-10-16.