

Présentation du Programme et des Objectifs

2024-02-13

Informatique de Gestion II

A. Larhlimi

ENCGS

13 Février 2024

Cours Tp

- Cycle Initial
- Informatique II
- soit 125 heures de travail, donc 2 heures de travail personnel pour 1 heure de cours
- 14 sessions

Intervenants

- A. Larhlimi

Évaluation

- contrôle continu (40%)
- devoir Projet (20%)
- contrôle final (40%)

Les devoirs et contrôles finaux auront lieu en ligne.

Objectifs du cours

Objectifs généraux du programme informatique de gestion :

##1. Améliorer l'efficacité et la productivité des organisations:

Automatiser les tâches répétitives et chronophages pour libérer du temps pour des activités plus stratégiques. Rationaliser les processus et éliminer les goulots d'étranglement pour améliorer la fluidité des opérations. Centraliser et organiser les données pour une meilleure accessibilité et une prise de décision plus éclairée. Améliorer la communication et la collaboration entre les équipes et les départements. ##2. Accroître la prise de décision stratégique:

Fournir des données précises et exploitables pour éclairer les décisions stratégiques. Développer des outils d'analyse et de reporting pour identifier les tendances et les opportunités. Simuler l'impact de différentes décisions avant de les mettre en œuvre. Améliorer la gestion des risques en identifiant et en atténuant les menaces potentielles. ##3. Renforcer la compétitivité des organisations:

Développer de nouveaux produits et services grâce à l'innovation technologique. Améliorer la satisfaction client en offrant une expérience plus fluide et personnalisée. Réduire les coûts en optimisant les processus et

en rationalisant l'utilisation des ressources. Gagner en agilité pour s'adapter rapidement aux changements du marché. ## Objectifs spécifiques du programme informatique de gestion : ##1. Compétences techniques:

Maîtriser les outils bureautiques (Word, Excel, PowerPoint, etc.). Comprendre les principes de base de la programmation et du développement web. Savoir administrer et gérer un système d'information. Être capable de déployer et de configurer des logiciels. Assurer la sécurité des données et des systèmes informatiques. ##2. Compétences analytiques:

Collecter et analyser des données quantitatives et qualitatives. Identifier les problèmes et proposer des solutions. Développer des outils de reporting et de visualisation de données. Maîtriser les techniques de statistique et de data mining. ##3. Compétences managériales:

Gérer des projets informatiques de bout en bout. Travailler en équipe et collaborer avec des experts d'autres domaines. Communiquer efficacement avec les utilisateurs et les parties prenantes. Comprendre les enjeux économiques et organisationnels de l'informatique. #4. Compétences transversales:

Avoir une bonne capacité d'adaptation et de résolution de problèmes. Être capable de travailler de manière autonome et en équipe. Avoir un bon esprit d'analyse et de synthèse. Savoir communiquer efficacement à l'oral et à l'écrit.

Informatique

Comment définir l'informatique ?

Un enjeu majeur de Société...

L'informatique et le Langage Python

Ouvrages de référence du Cours

Niveau Avancé

- Casamayou-Boucau et al. Programmation en Python pour les mathématiques. Dunod (2016)
- McKinney W. Analyse de données en Python, Manipulation de données avec pandas, NumPy et IPython. Eyrolles (2015)
- Chazallet. Python 3, Les fondamentaux du langage. Ressources informatiques (2014)
- Cormen et al. Algorithmique, Cours avec 957 exercices et 158 problèmes. Dunod (2010)
- Lutz et Ascher. Introduction to Python. O'Reilly (2000)

L'informatique et le Langage Python

Pour les débutants

Pour prendre le bon train

- Monk. Programming the Raspberry Pi, Getting started with Python. Mc Graw Hill (2013)
- Eric Grimson, Introduction to Computer Science and Programming using Python, MITx:6.00.1x (2016)
- Nantais et Downey. Think Python How to Think Like a Computer Scientist, Version 1.1.20. Green Tea Press (2013)

Informatique et Programmation: définition

- **informatique** = **information** + **automatique**

Objectifs

- Les deux types de connaissance nécessaires pour résoudre des problèmes
- Quelles sont les étapes que l'ordinateur doit suivre pour **executer** une tâche ?
- Quelle est la syntaxe que doit suivre le **langage** pour transcrire nos idées en étapes mécaniques

Algorithme : définition

- algorithme := suite finie et non ambiguë d'opérations élémentaires permettant de résoudre un problème
- opération élémentaire = **une instruction**
- **méthode de calcul pas à pas, que l'on peut automatiser avec une machine**
- le mot algorithme vient du nom du mathématicien perse Al-Khawarizmi (780 - 850), qui est également à l'origine du mot algèbre et de l'utilisation des chiffres arabes

Algorithmes et ordinateurs

- les algorithmes ont existé avant les ordinateurs
- cependant certains algorithmes ne sont pas simples à appliquer/exécuter manuellement car
 - leur temps d'exécution est trop long pour une personne
 - la probabilité d'une erreur humaine est trop grande (précision, répétition, ...)
- alors que les ordinateurs peuvent exécuter
 - très rapidement des instructions
 - de manière fiable
 - sans erreur

L'ordinateur est une machine Logique :

relation between expressions	meaning
(x) and (y)	SIMULTANEOUSLY both True
not (x)	the negation of x is True
(x) or (y)	INDEPENDENTLY True

Exemples

“Computational Thinking”

Plusieurs manières de faire Hello World!

```
print('Hello world !')
```

```
echo 'Hello world !'
```

```
5 int main(int argc, char *argv[]) {
6   cout << "Hello world ! \n" << endl;
7   return 0;
8 }
```

Un problème de connaissance

pour résoudre un problème par le calcul, il faut des opérations fondamentales

- Une connaissance des **déclarations** qui modélisent les aspects pertinents/essentiels du problème
- Une connaissance des **méthodes systématiques**, les points sur lesquels il faut s'appuyer

Premier exemple : somme des nombres (Problème)

On cherche à calculer la somme des nombres de 1 à n :

$$\sum_{k=1}^n k$$

- Que vaut la somme des nombres de 1 à 3 ?
- Que vaut la somme des nombres de 1 à 100 ?

Premier exemple : somme des nombres (Solution)

Au moins 3 manières de faire :

- commencer par les nombres les plus élevés
- commencer par les nombres les plus petits
- appliquer la formule $\frac{n*(n+1)}{2}$

Langages utilisés pour le cours

- Python : <https://docs.python.org/3/tutorial/>
- Blockly : <https://code.google.com/p/blockly>

Blockly

- langage de programmation visuel
- intégré au navigateur web
- pour découvrir les notions de base de programmation

Python

- créé par Guido van Rossum en 1989 (première version écrite en une semaine)
- multi paradigmes : impératif, fonctionnel, objet
- à typage dynamique
- simple à apprendre
- très expressif : plus proche du cerveau que du processeur
- très populaire
- utilisé par de grandes entreprises : google, pinterest, dropbox, aldebaran pour la programmation de Nao, ...
- multi plateformes (W\$, MacOS X, iOS, linux, ...)

Nous présenterons plusieurs algorithmes simples :

- Identification de Palindromes
- Calcul d'une racine carrée
- Détermination des années bissextiles

Des citations...

Celle-ci est de *Mitch Resnick*.

If you learn to read, you can then read to learn.
If you learn to code, you can then code to learn.