Révisions – Théorie - Programmation

**Logigrammes**

1. Quel est le symbole utilisé pour représenter une condition dans un logigramme ?

a) Un rectangle b) Un losange c) Un cercle d) Une flèche

1. Vrai ou Faux :

Un logigramme peut avoir plusieurs points de départ.

1. Expliquez la différence entre les structures linéaire, alternative et répétitive dans un logigramme.

Donnez un exemple d’algorithme pour chaque structure.

1. Pourquoi est-il utile de créer un logigramme avant d’écrire le code d’un programme ?

**Opérateurs logiques**

1. Complétez le tableau de vérité pour l’opérateur logique AND :



1. Que signifie l’opérateur logique OR ? Donnez un exemple de son utilisation en programmation.
2. Expliquez comment l’opérateur NOT modifie la valeur d’une expression booléenne.

**Opérateurs de comparaison**

1. Quels sont les opérateurs de comparaison utilisés en Python ? Donnez un exemple pour chaque opérateur.
2. Quelle est la différence entre l’opérateur == et l’opérateur = ?
3. Écrivez une structure conditionnelle qui vérifie si une variable age est supérieure ou égale à 18 et inférieure à 30. Si la condition est positive écrire « ok » sinon écrire « ko ».

**Python :**

**Chapitre 1 : Affichage de textes**

1. **Vrai ou Faux**: La fonction print() peut seulement afficher du texte entre guillemets. Expliquez votre réponse.

 **Chapitre 2 : Répétitions d'instructions**

1. **Comparer et contraster** : Quelle est la différence entre une boucle for et une boucle while ? Expliquez brièvement. Donnez un exemple pour chacune des boucles qui affiche les nombres de 1 à 5.
2. **Analyse de code**: Examinez le code suivant et expliquez ce qu'il fait :

for i in range(5):

 for j in range(i+1):

 print("\*", end="")

 print()

1. **Erreur d'indentation**: Expliquez pourquoi l'indentation est cruciale en Python, en particulier dans le contexte des boucles. Donnez un exemple de code avec une erreur d'indentation et expliquez le problème.

**Chapitre 3 : Calculs et Variables**

1. **Types de données**: Associez chaque valeur à son type de données en Python :
	* "Bonjour" => ...
	* 123 => ...
	* 3.14 => ...
	* True => ...
2. **Priorité des opérateurs**: Évaluez l'expression suivante en respectant la priorité des opérateurs : 10 - 2 \* 3 + 5 / 2. Détaillez les étapes de calcul.
3. **Définition**: Expliquez ce qu'est une variable et comment elle est utilisée en programmation. Donnez des exemples de noms de variables valides et invalides en Python.

**Chapitre 4 : Les entrées et les conditions**

1. **Conversion de type**: Pourquoi est-il parfois nécessaire d'utiliser int() avec la fonction input() ? Donnez un exemple concret.
2. **Structures conditionnelles**: Expliquez la différence entre les mots-clés if et else. Donnez un exemple de code qui utilise les deux.
3. **Opérateurs de comparaison**: Listez tous les opérateurs de comparaison en Python et expliquez leur signification avec des exemples.

**Chapitre 5 : La boucle while et les fonctions**

1. **Boucle infinie**: Qu'est-ce qu'une boucle infinie et comment peut-elle se produire avec une boucle while ? Donnez un exemple de code.
2. **Avantages des fonctions**: Expliquez les avantages d'utiliser des fonctions dans un programme.