

# Algorithmie – Exercices

## Structures de contrôle SI-SINON

### Table des matières

1	Consignes générales .....	2
2	Structure de contrôle SI-SINON.....	2
2.1	Exercice 8 – Facture.....	2
2.2	Exercice 9 – Heures supplémentaires .....	2
2.3	Exercice 10 – Plus petit nombre .....	2
2.4	Exercice 11 – Compte bancaire .....	2
2.5	Exercice 12 – Commission .....	3
2.6	Exercice 13 – Moyenne et cote .....	3
2.7	Exercice 14 – Relevé de compte.....	3
2.8	Exercice 15 – Rapport d’inventaire .....	4
2.9	Exercice 16 – Frais de scolarité.....	4
2.10	Exercice 17 – Relevé de compte 2.....	4
2.11	Exercice 18 – Vérification taille et poids .....	4
2.12	Exercice 19 – Rapport de main d’œuvre forfaitaire .....	4
2.13	Exercice 20 – Moyenne et note lettrée .....	5
2.14	Exercice 21 – Candidat à un poste.....	5

### 1 Consignes générales

Vous devez implémenter avec le logiciel LARP en pseudo-code les algorithmes définis dans chacun des énoncés suivants. Vous veillerez à vérifier le bon fonctionnement de vos algorithmes en les testant avec des valeurs spécifiques dont vous aurez déterminer le résultat attendu.

Chaque algorithme donnera lieu à la création d'un fichier nommé avec le formalisme suivant : **pnom\_ex\_n°.larp** (exemple : mdupont\_ex\_2.larp). Vous déposerez sur FOCAL chacun des exercices dans la zone de dépôt correspondante afin de débloquent une correction.

### 2 Structure de contrôle SI-SINON

#### 2.1 Exercice 8 – Facture

Concevoir un algorithme calculant la facture d'un client. La facture se calcule en fonction du prix d'un article et de la quantité achetée :

- Le client obtient un rabais de 10% HT, seulement s'il achète pour plus de 200€ HT.
- Les taxes s'élèvent à 15% du prix d'achat.

L'algorithme doit lire le nom du client, l'article acheté, la quantité et le prix. Puis, il doit afficher le nom de ce client, l'article qu'il s'est procuré et le montant de sa facture.

#### 2.2 Exercice 9 – Heures supplémentaires

Concevoir un algorithme capable de déterminer si un employé a travaillé des heures supplémentaires :

- Un employé a travaillé des heures supplémentaires, seulement s'il a effectué plus de 40 heures de travail.

L'algorithme doit lire le numéro d'employé, le nom de l'employé et le nombre d'heures travaillées. Puis, il doit afficher toutes cette information, seulement si l'employé a travaillé des heures supplémentaires. Sinon, rien n'est affiché à l'écran.

#### 2.3 Exercice 10 – Plus petit nombre

Concevoir un algorithme capable de trouver le plus petit nombre d'une certaine série.

L'algorithme doit lire 4 nombres et afficher le plus petit d'entre eux.

#### 2.4 Exercice 11 – Compte bancaire

Concevoir un algorithme capable de déposer ou retirer de l'argent d'un compte bancaire selon le choix de l'utilisateur :

- Un code de transaction de 1, signifie que le client veut faire un dépôt.
- Un code de transaction de 2, signifie que le client veut faire un retrait.
- Tout autre code de transaction représente une erreur d'entrée du client.

L'algorithme doit lire le nom du client, le numéro du compte, le solde initial, le montant de la transaction et un code de transaction. Puis, il doit afficher le nom du client, le numéro de compte, le solde initial et final du compte.

### 2.5 Exercice 12 – Commission

Concevoir un algorithme calculant la commission que recevra un employé en fonction de la classe d'employé à laquelle il appartient :

- **Classe 1 :**
  - Si le total de vente de l'employé est plus petit ou égal à 1000€, le taux de commission est de 6%.
  - Si le total de vente de l'employé est plus grand que 1000€, mais plus petit ou égal à 2000€, le taux de commission est de 7%.
  - Si le total de vente de l'employé est plus grand que 2000€, le taux de commission est de 10%.
- **Classe 2 :**
  - Si le total de vente de l'employé est plus petit que 1000€, le taux de commission est de 4%.
  - Si le total de vente de l'employé est plus grand ou égal à 1000€, le taux de commission est de 6%.
- **Classe 3 :**
  - Le taux de commission est de 4.5% peu importe le total des ventes.
- **Classe 4 :**
  - Le taux de commission est de 5% peu importe le total des ventes.
- Tout autre classe d'employé représente une erreur d'entrée de l'utilisateur.

L'algorithme doit lire le nom, le numéro, le total des ventes et la classe de l'employé. Puis, il doit afficher le nom, le numéro et la commission de l'employé seulement si la classe est valide.

### 2.6 Exercice 13 – Moyenne et cote

Concevoir un algorithme calculant la moyenne de 3 notes d'un étudiant et qui détermine la cote de l'étudiant :

- L'étudiant se mérite la cote 'S', si sa moyenne est égale ou supérieur à 65%.
- L'étudiant obtient la cote 'U', si sa moyenne est inférieure à 65%.

L'algorithme doit lire le nom et le numéro de l'étudiant, ainsi que 3 notes. Puis, il doit afficher le nom et la cote de l'étudiant. Si la cote est 'U', on affiche aussi le nombre de points manquants pour obtenir la cote 'S'.

### 2.7 Exercice 14 – Relevé de compte

Concevoir un algorithme capable de produire un relevé de compte mensuel pour un client faisant usage d'une carte de crédit. L'algorithme prend en compte le total des achats et des paiements effectués par le client afin de calculer le montant d'intérêts et ensuite le solde final du compte :

- Le montant soumis à des frais financiers (intérêts), se calcule en additionnant le total des achats et des paiements au solde initial du compte.
- Si le montant soumis à des frais financiers est plus grand ou égal à 250€, le taux d'intérêt est de 1.5%.
- Si le montant soumis à des frais financiers est plus petit que 250€, le taux d'intérêt est de 1%.

- Le solde final est calculé en additionnant le montant soumis à des frais financiers et le montant des intérêts.

L'algorithme doit lire le nom du client, le solde initial du compte et le total des achats et le total des paiements. Le relevé de compte mensuel qui sera affiché, doit contenir le nom du client, le solde initial, le total des achats et des paiements, le montant soumis à des frais financiers, le montant des intérêts et le solde final du compte.

### 2.8 Exercice 15 – Rapport d'inventaire

Concevoir un algorithme capable de produire un rapport d'inventaire pour un certain article. La valeur d'inventaire est calculée en fonction de la quantité en inventaire et du prix unitaire de l'article.

L'algorithme doit lire le numéro de l'article, la quantité en inventaire et le prix unitaire. Le rapport d'inventaire qui sera affiché, doit contenir le numéro de l'article et la valeur d'inventaire ou le montant excédant 1000€ de valeur d'inventaire, dépendamment de la valeur d'inventaire.

### 2.9 Exercice 16 – Frais de scolarité

Concevoir un algorithme capable de produire un rapport de facturation scolaire d'un étudiant. Les frais de scolarité sont calculés en fonction du nombre de crédits auxquels est inscrit l'étudiant :

- Si l'étudiant est inscrit à au moins 10 crédits, les frais de scolarité sont fixes à 1000€.
- Si l'étudiant est inscrit à moins de 10 crédits, les frais de scolarité sont de 100€ par crédits.

L'algorithme doit lire le nom et le Numéro de Sécurité Sociale de l'étudiant, ainsi que le nombre de crédits auxquels il est inscrit. Le rapport de facturation scolaire qui sera affiché, doit comprendre le nom, le Numéro d'Assurance Sociale et les frais de scolarité de l'étudiant.

### 2.10 Exercice 17 – Relevé de compte 2

Concevoir un algorithme capable de produire un relevé de compte mensuel pour un certain compte de banque :

- Si le solde final à la fin du mois est plus petit que 100€, des frais de service de 5€ sont soustraits au solde.

L'algorithme doit lire le nom et le numéro de compte du client, le solde initial, le total des dépôts et des retraits effectués au courant du mois. Le relevé de compte qui sera affiché, doit comprendre le nom et le numéro de compte du client, ainsi que le solde initial et final du compte.

### 2.11 Exercice 18 – Vérification taille et poids

Concevoir un algorithme déterminant si une personne est, en même temps, plus grande que 1.52 mètres et si elle pèse plus de 45 kilogrammes :

L'algorithme doit lire le nom, la grandeur (en mètres) et le poids (en kilogrammes) d'une certaine personne, puis afficher ou non le nom de la personne selon sa grandeur et son poids.

### 2.12 Exercice 19 – Rapport de main d'œuvre forfaitaire

Concevoir un algorithme capable de produire un rapport de main-d'œuvre forfaitaire. Le salaire de chaque ouvrier est calculé en fonction de la qualification de chacun :



- Un ouvrier qualifié a un code d'emploi de « Q » et un salaire de 12.00€.
- Un ouvrier apprenti a un code d'emploi de « A » et un salaire de 10.00€.
- Un ouvrier stagiaire a un code d'emploi de « S » et un salaire de 8.00€.

L'algorithme doit lire le nom d'un ouvrier, la position occupée, le nombre d'heures travaillées par jour et un code. Le rapport de main-d'œuvre forfaitaire qui sera affiché, doit comprendre le nom, la position occupée, les heures travaillées et le montant de la paye.

### 2.13 Exercice 20 – Moyenne et note lettrée

Concevoir un algorithme calculant la moyenne d'un étudiant en fonction de 3 notes d'examen et qui détermine ensuite la note en lettre. Pour déterminer la note en lettre, on se base sur le tableau suivant :

Moyenne	Note en lettre
90 à 100	A
80 à 89	B
70 à 79	C
60 à 69	D
50 à 59	E
0 à 49	F

L'algorithme doit lire le nom de l'étudiant et 3 notes (sur 100%), puis afficher note en lettre correspondant à la moyenne.

La sortie doit ressembler à ceci : \_\_\_\_\_ s'est mérité(e) la note \_\_\_\_ .

### 2.14 Exercice 21 – Candidat à un poste

Concevoir un algorithme capable de déterminer si un postulant est éligible pour un certain poste. Pour être éligible, un candidat doit répondre à chacun de ces critères :

- Le candidat doit être un citoyen des Etats-Unis.
- Le candidat doit être âgé d'au moins 21 ans.
- Un code est utilisé pour déterminer la citoyenneté d'un candidat. Si ce code est 1, le candidat est un citoyen des États-Unis. Si ce code est 2, il n'est pas un citoyen des États-Unis.

L'algorithme doit lire le nom et l'âge du candidat, ainsi qu'un code correspondant à sa citoyenneté. Puis, il affiche un message spécifiant lequel ou lesquels des tests qui n'ont pas été passés, ou bien, si le candidat répond à tous les critères, il n'affiche que le nom du candidat.