Mini Projet 1

Implanter un algorithme de compression qui transforme une chaine de caractères de type

aaaaabbfffffff en a5b2f7 et qui évalue le pourcentage de compression

Mini Projet 1bis

Implanter un algorithme de décompression qui transforme une chaine de caractères de type

a5b2f7 en aaaaabbfffffff

Mini Projet 2

Le distributeur de boisson

**Description**

Le but de ce projet est de simuler le fonctionnement d’un distributeur de boissons. Chaque produit de distributeur possède un nom, une quantité en stock et un prix (en centimes). Le distributeur contient en plus des pièces de monnaie dans son monnayeur. Chaque pièce de monnaie contient une valeur en centimes (5,10,20,50,100 ou 200) et une quantité disponible. Pour faire son choix, le consommateur visualise le distributeur virtuel sous la forme d’une liste de produits disponibles. Un produit (ex : ‘Barre céréales’) est associé à un code dans la machine (ex : le code 81), vaut un certain prix et est disponible avec une certaine quantité (un produit indisponible a une quantité à 0). L’état des pièces contenues dans la machine est également affiché.

Initialement l'état du distributeur et du monnayeur est affiché, suivi d'un menu qui propose les choix suivants :

1. **fin** : le programme se termine.
2. **choisir un produit** : l’utilisateur entre le code d’un produit.
3. **entrer des pièces** : l’utilisateur entre successivement une ou plusieurs pièces de monnaie. Les pièces entrées par l’utilisateur sont affichées.
4. **obtenir le produit choisi** (et obtenir sa monnaie) : Pour obtenir le produit choisi, il faut que le code du produit soit valide et que la somme totale des pièces rentrées soit suffisante par rapport au prix du produit. Pour simuler le produit qui tombe dans le distributeur, afficher un message contenant l’information sur le produit acheté. Le programme calcule aussi le tableau des pièces à rendre au consommateur en fonction du prix du produit, de l’argent déjà mis dans le distributeur et des pièces disponibles. Il affiche la monnaie rendue.
5. **prendre la monnaie** : la monnaie disparaît.
6. **remboursement** : si l’utilisateur constate qu’il n’a pas assez d’argent ou s’il a tout simplement changé d’avis, il peut demander le remboursement des pièces déjà insérées.
7. **recharger(Avancé)** : la maintenance recharge le distributeur en indiquant le nombre supplémentaires insérés dans le distributeur pour chaque produit disponible. La caisse est aussi vidée.

Votre travail: Essayer au minimum de faire la première question.

1) Vous écrirez d'abord l'algorithme qui rend la monnaie en supposant que le monnayeur contient toujours suffisamment de chaque pièce et qu'il est possible de rendre la monnaie.(faites une fonction).

2) Vous corrigerez votre algorithme en intégrant le fait que le monnayeur se vide et se remplis.

3) Vous faites l'interface.

Mini projet 3

Manipulation de polynômes

**Description**

**Objectif** : Le but de ce projet est de définir un ensemble de fonctions capables de manipuler des polynômes.

**Notions utilisées** : Programmation

**Contexte** : Un polynôme à une variable est un objet mathématique que l'on définit comme étant la somme des termes P (x) = a\_0 + a\_1 x + a\_2 x^2 + . . . + a\_n x^n . Chaque terme du polynôme est de la forme a\_k x^k dans lequel x est la variable du polynôme, k le degré du terme et a\_k son coefficient. On appelle alors degré du polynôme le plus grand degré de ses termes.

Ainsi, un polynôme de degré n comme P (x) = a\_0 + a\_1 x + a\_2 x^2 + . . . + a\_n x^n est complètement défini par la liste des coefficients (a\_0 a\_1 … a\_n ). Par exemple, le polynôme P(x) = 14 + x − 4x^2, de degré 2 est déterminé par la liste de ses coefficients (14 1 -4). De même, le polynôme P(x) = x + 3 x^2 + 5 x^7 , de degré 7, est représenté par (0 1 3 0 0 0 0 5).

**Ce qu'il faut réaliser** :

* Implémenter une bibliothèque permettant de manipuler les polynômes

**Quelques pistes de reflexion** :

Ci dessous une liste non exhaustive de fonctions auxquelles on peut penser :

* Calculer le degré d'un polynôme
* Calculer la valeur en x d'un polynôme
* Tester si un nombre est racine d'un polynôme
* Effectuer l'addition de deux polynômes
* Effectuer la multiplication de deux polynômes

Mini projet 4

Manipulation de polynômes

**Description**

**Objectif** : Le but de ce projet est de définir un ensemble de fonctions capables de manipuler des matrices.

**Notions utilisées** : Programmation

**Contexte** : Une matrice de taille (n,p) désigne une collection de valeurs indexées par les couples (i,j) (avec i ⇐ n et j ⇐ n) que l'on représente généralement par un tableau rectangulaire. Nous représentons ici les matrices par un tableau de tableaux. Une matrice est donc un tableau de n éléments, chaque élément étant lui même un tableau de p éléments.

Le but de ce projet est de proposer un ensemble de fonctions permettant la création et la manipulation de matrices.

**Ce qu'il faut réaliser** :

* Implémenter une bibliothèque permettant de manipuler les matrices.

**Quelques pistes de réflexion** :

Ci dessous une liste non exhaustive de fonctions auxquelles on peut penser :

* Création d'une matrice nulle de taille (n,p)
* Création d'une matrice identité de taille n
* Création d'une matrice diagonale
* multiplication d'une matrice par un nombre
* Effectuer l'addition de deux matrices
* Effectuer la multiplication de deux matrices (combien de multiplications sont faites?)
* Calcul de la transposé d'une matrice
* Tester si une matrice est symétrique

Mini projet 5

Le jeu des allumettes

## Description

**Objectif** : Réfléchir à des structures de données adaptées au jeu des allumettes et proposer une implémentation permettant au joueur de jouer contre un autre joueur ou contre la machine.

**Contexte** : Le jeu des allumettes (ou jeu de Nim) est un jeu à deux joueurs très simple dans lequel on dispose d'un ensemble d'allumettes alignées côte à côte. Chaque joueur retire à tour de rôle du plateau de jeu 1, 2 ou 3 allumettes placées consécutivement. Le joueur qui retire la dernière allumette a perdu.

Le but du projet est de conduire une réflexion sur une représentation adaptée de ce jeu en machine afin de permettre à 2 joueurs (humains) de s'affronter.

Il faut analyser toutes les taches à accomplir et créer des fonctions pour chaque tache.

Dans un second temps, il est demandé de réfléchir à différentes stratégies à implémenter pour permettre à un joueur d'affronter l'ordinateur

Mini projet 6

a) Ecrivez un programme qui remplace chaque voyelle dans une chaine de caractéres par la voyelle suivante ( a devient e, e devient i, i et y deviennent o, o devient u et u devient a)

b) modifiez votre programme pour qu'il écrive en "javanais": on rajoute "av" avant chaque voyelle ou groupe de voyelles (par exemple " bonjour madame" devient "bavonjavour mavadavamave")

Mini projet 6bis

pour écrire en "javanais": on rajoute "av" avant chaque voyelle ou groupe de voyelles (par exemple " bonjour madame" devient "bavonjavour mavadavamave")

Ecrire un programme qui décrypte une phrase en javanais

Mini projet 7

Jules César utilisait un code cryptographique pour communiquer avec ses lieutenants: il remplaçait chaque lettre par son successeur dans l'alphabet: A devient B, B devient C, ..., Y devient Z et Z devient A

a) Ecrivez un programme qui code puis décode une phrase quelconque

b) Une amélioration consiste à décaler chaque lettre de n emplacements. Réutilisez l'algorithme précédent n fois.

Mini projet 8

Ecrivez un programme qui calcule les termes de la suite de Syracuse:

Remarquez que pour n'importe quel nombre de départ a, la suite se termine à 1.

On appelle VOL d'un nombre a, le nombre de termes utilisés avant d'arriver à 1 et HAUTEUR de VOL la valeur maximale atteinte.

On stockera dans des listes les longueurs et hauteurs de vol pour chaque valeur de a entier ≤1000 et on affichera des indicateurs numériques qui permettent de faire des commentaires .