



DESCRIPTION D'UNE SITUATION PROFESSIONNELLE

PARCOURS	SISR <input type="checkbox"/>	SLAM <input checked="" type="checkbox"/>
Lieu de réalisation	CFA CERFAL Campus Montsouris Paris 14ème	 
Période de réalisation	Du : 27/11/22	Au : 17/01/23
Modalité de réalisation	SEUL <input checked="" type="checkbox"/>	<input type="checkbox"/>

Intitulé de la mission	Développement d'un programme de gestion de tableau
Description du contexte de la mission	Projet à effectuer seul en formation : concevoir et développer un programme permettant à l'utilisateur de gérer un tableau en effectuant différentes opérations telles que l'initialisation, l'affichage, la modification et la suppression des valeurs du tableau. Une interface graphique sera également créée pour afficher le tableau et les options de gestion.

Contraintes & Résultat	Ressources fournies / contraintes techniques / Résultats attendu
	<p><u>Contraintes</u> : Délais de développement, les ressources disponibles, les spécifications techniques du programme, ainsi que les attentes des utilisateurs en termes de fonctionnalités.</p> <p><u>Résultat attendu</u> : Créer un programme en C# permettant de gérer un tableau à 1 ou 2 dimensions en offrant des options d'initialisation, d'affichage, de dédoublonnage, de suppression et d'ajout de valeurs</p>
Productions associées	Liste des documents produits et description
	Code source du projet final en C# réalisé sur Visual Studio

Modalités d'accès aux productions	Identifiants, mots de passe, URL d'un espace de stockage et présentation de l'organisation du stockage
	Code source via un éditeur de texte

Description détaillée de la situation professionnelle retenue et des productions réalisées en mettant en évidence la démarche suivie, les méthodes et les techniques utilisées

- Création des différentes variables selon les dimensions, la taille et les méthodes à produire : initialisation des variables nécessaires pour la gestion du tableau, variables pour stocker les dimensions du tableau, la taille des éléments, les index de ligne et de colonne, ainsi que d'autres variables nécessaires pour les opérations de gestion telles que l'ajout, la suppression et la déduplication de valeurs.

- Création des méthodes pour initialiser et afficher le tableau, dédoubler, supprimer et ajouter des valeurs :

Mise en place de différentes méthodes pour effectuer les opérations de gestion du tableau. Une méthode d'initialisation permettra de créer un tableau vide ou de le remplir avec des valeurs par défaut. Une méthode d'affichage sera développée pour présenter le tableau à l'utilisateur. Des méthodes spécifiques seront implémentées pour dédoubler les valeurs, supprimer des éléments et ajouter de nouvelles valeurs dans le tableau, en tenant compte des dimensions et de la taille spécifiées.

- Création de l'interface graphique pour afficher le tableau et les options de gestion du tableau : Une interface graphique conviviale sera conçue pour permettre à l'utilisateur d'interagir avec le tableau. Cela inclura la création d'une fenêtre ou d'une zone d'affichage où le tableau sera représenté visuellement.

- Création d'un menu pour permettre à l'utilisateur d'interagir avec le tableau en sélectionnant une case :

Un menu intuitif sera développé pour permettre à l'utilisateur de sélectionner une case spécifique du tableau et d'effectuer des actions sur cette case.

- Réalisation de tests pour s'assurer que chaque méthode de gestion du tableau fonctionne correctement séparément :

Tests et vérification du bon fonctionnement de chaque méthode de gestion du tableau de manière isolée afin de s'assurer que chaque méthode effectue les opérations attendues, respecte les contraintes de dimension et de taille, et produit les résultats attendus.

- Intégration des différentes méthodes pour vérifier que l'ensemble fonctionne de manière cohérente :

Intégration des méthodes pour former un ensemble cohérent. Cela implique de s'assurer que les différentes méthodes interagissent correctement entre elles, en respectant l'ordre d'exécution et en utilisant les valeurs appropriées.

- Correction des éventuels bugs ou problèmes si nécessaire :

Mesures correctives prises pour résoudre ces problèmes telles que la modification du code, l'ajustement des paramètres, l'optimisation des performances. L'objectif est de garantir que le programme fonctionne de manière fiable et sans problème.

```

void triDecroissant2D(ref int[,] Tablo, int TailleEnLignes, int TailleEnColonnes)
{
    int nb=0;
    for (int i = 0; i < TailleEnLignes; i++)
    {
        for (int j = 0; j < TailleEnColonnes; j++)
        {
            for (int m = i; m < TailleEnLignes; m++)
            {
                for (int n = j; n < TailleEnColonnes; n++)
                {
                    if (Tablo[m, n] > Tablo[i, j])
                    {
                        nb = Tablo[m, n];
                        Tablo[m, n] = Tablo[i, j];
                        Tablo[i, j] = nb;
                    }
                }
            }
        }
    }
}

```

----- A R R A Y P R O G -----

Commençons par créer un nouveau tableau.

Quelle dimension souhaitez-vous pour le tableau ?

[1] 1 Dimension

[2] 2 Dimensions

Votre choix : 2

Nombre de Lignes : 2

Nombre de Colonnes : 2

[D] - Trier le tableau :

[c] - Par ordre croissant

[d] - Par ordre décroissant

Votre choix : c

Le tableau a été trié par ordre croissant.

[C] - Afficher le contenu du tableau :

1	5	8	12	23
3	20	24	31	38
11	30	39	51	54
14	41	49	58	77
38	54	86	90	92

[G] - Supprimer une valeur du tableau

[Q] - Quitter le menu

Votre choix :