

TD6 - Algorithmique et C - Matrices

1 Schéma d'exécution

EXERCICE 1 *Pour chacun des deux codes ci-dessus, faire le schéma d'exécution :*

<pre>int algo1(int a, int b){ int tmp = a+2*b return temp } int main(){ int a, int j; j=34 ; a=2; int resu = algo1(j,a); return 0; }</pre>	<pre>void algo2(char* s,int b){ b = b+1; *s = 'c'; } int main(){ int x=1; char u; algo2(&u,x); printf("%c, %d",u,x); }</pre>
---	--

2 Matrices

Premiers exos simples On écrira directement les programmes en C

EXERCICE 2 *Déclarer un tableau à 100 lignes et 200 colonnes, et en utilisant la fonction `hasard`, initialiser le tableau avec des entiers tirés au hasard entre 0 et 50.*

EXERCICE 3 *Écrire une procédure `affiche_2d` qui prend en paramètre un tableau à CMAX colonnes et LMAX lignes et qui affiche ce tableau en lignes et colonnes.*

Analyse/Conception

EXERCICE 4 *Écrire un algorithme `somme_tab2dim` qui prend en paramère un tableau `tab` à deux dimensions et qui :*

- additionne, colonne par colonne, les lignes du tableau et place le résultat dans un tableau ligne;
- idem par ligne
- additionne toutes les cases.

EXERCICE 5 *Source : FS pour GIS*

Soient n un entier et $M[n][n]$ une matrice d'entiers. Déterminer l'indice de la colonne dont la somme des coefficients est la plus grande.

EXERCICE 6 *(même source) Soient n un entier et $M[n][n]$ une matrice de booléens représentant une relation R sur les entiers de la façon suivante : $M[i][j] = \text{VRAI}$ si et seulement si iRj . Écrire des algorithmes permettant de déterminer si R est réflexive, symétrique, transitive.*