

NOM:	PRENOM:	DIVISION
------	---------	----------

COMPOSITION D'INFORMATIQUE 2-2**(Divisions A, B , C , D)**

Durée : 2 heures

Epreuve sans document

Répondre sur la feuille

1) Questions sur les boucles*a. Tests sur les entiers*

Ecrire un main(), contenant une boucle for(), écrivant les N premiers nombres entiers strictement positifs qui ne sont divisibles par aucun des nombres 2, 5, ou 7. En voici une exécution

```
% gcc pgm.c -o pgm -Wall
% ./pgm
Valeur de N ? 10
1 .. 3 .. 9 .. 11 .. 13 .. 17 .. 19 .. 23 .. 27 .. 29 ..
%
```

#include

..... main(.....) {

int N,i,written;

printf ("Valeur de N ? ");

..... // Saisie de la valeur de N

for(.....)**if**(.....)

printf(")

.....

.....// Ecrire le retour à la ligne sans utiliser printf()

.....

}

Réécrire la partie du programme contenant le test d'écriture sur la console, en utilisant la condition contraire. On demande ici la solution avec un else, moins élégante que la précédente.

```
for( ..... )  
    if( ..... )  
        .....  
    .....  
    .....  
    .....
```

Y a-t-il une solution à cette dernière question avec le test contraire, mais sans utiliser else ?

.....

b. *Fonction de hasard*

Comment écrire une liste pseudo-aléatoire de 100 nombres, qui change selon le moment où on lance l'exécution du programme ?

```
#include .....  
#include .....  
#include .....  
..... main( ..... ) {  
    int compteur;  
    .....  
    do  
        .....  
        .....  
        .....  
        .....  
}
```

Donner trois écritures différentes de l'incréméntation du compteur

.....

2) Questions sur les pointeurs*a. Allocation dynamique*

Par quelle instruction créer **dynamiquement** un tableau V de 100 entiers ?

int

Comment libérer la mémoire ainsi allouée ?

.....

Quelle bibliothèque contient les deux fonctions utilisables possibles? Quelle différence y a-t-il entre elles ?

.....

.....

b. Changer le contenu d'une variable dans une fonction

On fait un programme pour tester les différentes manières de gérer le calcul du milieu de deux points. Pour cela on envisage trois méthodes :

- Deux prototypes de fonctions qui, partant de la connaissance de deux points A et B, changent les coordonnées d'un troisième point M en y inscrivant les coordonnées du milieu.
- Une macro à paramètres SET_MIDPOINT(A,B,M)

Ecrire la définition d'une structure Point contenant deux champs x et y de type int.

typedef

.....

.....

.....

Ecrire les prototypes et un exemple d'appel de cette fonction depuis une autre fonction.

// Premier prototype contenant deux arguments

.....

Exemple d'appel :

// Second prototype contenant trois arguments

.....

Exemple d'appel :

Ecrire la macro et un exemple d'utilisation de cette macro dans une fonction.

.....

Exemple d'utilisation :

c. Liste chaînée

Une liste chaînée modélise un polygone non fermée (i.e une ligne brisée) et on veut calculer la longueur totale de cette ligne.

On suppose la structure Point établie comme à la question précédente.

Ecrire la définition d'une structure **Element** permettant de faire une liste chaînée de points.

typedef

.....

.....

.....

La bibliothèque math.h propose la fonction **double hypot(double x, double y);** qui renvoie la valeur $\sqrt{x^2 + y^2}$.

Ecrire le code d'une fonction qui renvoie la longueur de la liste chaînée. Un bonus sera attribué aux solutions qui utilisent un minimum de variable(s) locale(s).

..... **length**(.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Comment faire exécuter cette fonction depuis une autre fonction ?

.....

.....

d. Chaînes de caractères

Critiquer les codes et si il y a problème, proposer une solution plus adaptée :

```
char * s;  
strcpy(s,"Bonjour");  
puts(s);
```

.....

.....

.....

```
char * s="Bonjour";  
puts(s);  
strcpy(s,"Au revoir");  
puts(s);
```

.....

.....

.....

```
char s[5];
int i=2010;
printf("Vive les examens %d", i);
puts(s);
```

.....

.....

.....

.....

```
int main(void) {
    ...
    char * s=saisie();
    puts(s);
    ....
}

char * saisie(void) {
    char s[100];
    puts("Entrez une chaîne avec"
        "moins de 99 caractères");
    fgets(s,100,stdin);
    return s;
}
```

[illegible]

3) *Matrices*

Ecrire un programme comportant un `main()` et une fonction.

- Le `main()` demande deux entiers à l'utilisateur *nRows*, *nCols*.
- Le `main()` déclare ensuite une matrice de *nRows* lignes et *nCols* colonnes de type double.
- Le `main()` initialise cette matrice selon la règle : le coefficient en position (i,j) est l'inverse de (i+j+1).
- Le `main()` appelle la fonction qui se charge d'écrire la matrice sur la console, le formatage étant conforme à l'écriture ci-dessous.

```
% ./pgm
Nombre de lignes ? 3
Nombre de colonnes ? 4
1.00 | 0.50 | 0.33 | 0.25
0.50 | 0.33 | 0.25 | 0.20
0.33 | 0.25 | 0.20 | 0.17
%
```

*Modèle mathématique
que l'on veut coder en C*

$$\begin{pmatrix} \frac{1}{1} & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \end{pmatrix}$$

a. *Ecrire la partie entête du programme.*

.....

.....

.....

.....

.....

.....

.....

b. *Ecrire le `main()`.*

.....

.....

.....

.....

.....

.....

.....

c. Ecrire le code de la fonction.

.....

.....

.....

.....

.....

.....

.....

d. Variantes : que changer pour que les valeurs nRows et nCols soient lues sur la ligne de commande ?

```
% ./pgm 3 4
1.00 | 0.50 | 0.33 | 0.25
0.50 | 0.33 | 0.25 | 0.20
0.33 | 0.25 | 0.20 | 0.17
%
```

.....

.....

.....

.....

.....

.....

.....

e. Quelle instruction, d'un seul tenant, déclare et initialise une matrice à la matrice ci-contre ?

$$\begin{pmatrix} \frac{1}{1} & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \end{pmatrix}$$

.....

.....

.....

4) Structures & fichiers

Un fichier est destiné à recevoir les résultats de la première phase du Mondial de Football en suivant le modèle ci-dessous.

- A la création du fichier, les matchs sont inscrits sans score (exemple France Mexique).
- Au fur et à mesure les résultats sont inscrits (exemple Afrique_du_sud 1 12 France).

```
% cat Resultats_Mondial.txt
Groupe A
Afrique_Du_Sud 1 12 France
Afrique_Du_Sud 1 1 Mexique
Afrique_Du_Sud Uruguay
France Mexique
France 16 3 Uruguay
Mexique 7 9 Uruguay
Groupe B
etc ...
%
```

a. Une variable globale

En variable globale, on met en place une matrice de pointeurs permettant de stocker les pays de chaque groupe, et d'atteindre dans le programme un pays par la notation **pays[i][j]**, i représentant le groupe et j le rang dans le groupe.

1	<p> GROUPE A Afrique_du_Sud, Mexique, Uruguay, France GROUPE B Argentine, Nigeria, Coree_du_Sud, Grece GROUPE C Angleterre, Etats-Unis, Algerie, Slovenie GROUPE D Allemagne, Australie, Serbie, Ghana GROUPE E Pays-Bas, Danemark, Japon, Cameroun GROUPE F Italie, Paraguay, Nouvelle-Zelande, Slovaquie GROUPE G Bresil, Coree_du_Nord, Cote_d'Ivoire, Portugal GROUPE H Espagne, Suisse, Honduras, Chili </p>
---	--

..... pays =

.....

.....

..... *etc...*

b. La liste des groupes

Un `main()` écrit sur le terminal la liste des groupes selon la présentation 1. Donner la boucle :

```
int gr; // gr représente l'indice du groupe
```

```
for( ..... )
```

.....

.....

.....

c. Fonction pour inscrire un score

On suppose le fichier Resultats_Mondial.txt initialisé sur le modèle du premier encadré, sans score lorsque le match n'a pas été joué. Il s'agit de faire un programme permettant d'inscrire un résultat selon l'exécution présentée ci-contre :

Compléter sans indiquer les bibliothèques, ni la variable globale pays.

La fonction **strstr()** renvoie NULL si la chaîne *s2* n'est pas contenue dans la chaîne *s1* :

char * strstr(const char *s1, const char *s2);

La fonction **system()** permet de faire exécuter une commande depuis un programme.

```
% ./inscrireResultat
```

```
Donner le groupe ? F
```

```
Choisir les numéros des deux pays parmi
```

```
1 Italie
```

```
2 Paraguay
```

```
3 Nouvelle-Zelande
```

```
4 Slovaquie
```

```
? 4 2
```

```
Quel a été le résultat de ce match ? 2 3
```

```
Enregistrement dans le fichier de la ligne
```

```
Paraguay 3 2 Slovaquie
```

```
%
```

int miseAJourFichier(// ne pas renseigner le prototype ici, cela sera fait page suivante

```
..... main( ..... ) {
```

```
.....
.....
.....
/* saisie sécurisée du groupe (une lettre entre A et H, refuser les autres) */
.....
.....
.....
```

```
printf("Choisir les numéros des deux pays parmi\n");
```

```
/* afficher les pays du groupe choisi comme dans l'exemple */
.....
.....
.....
```

```
printf("? ");
```

```
/* saisie sécurisée des numéros des deux pays (entre 1 et 4, refuser les autres) */
.....
.....
.....
```

```
printf("Quel a été le résultat de ce match ? ");
.....
.....
```

```
int erreur;
```

```
...../* mise à jour du fichier */
```

```
if(erreur==0) {
```

```
    printf("Enregistrement dans le fichier de la ligne\n");
```

```
    printf( .....);
```

```
} else printf("ERREUR %d. L'enregistrement du résultat a échoué.\n",erreur);
```

```
return 0; }
```

```
int miseajourfichier(.....)
```

.....)

[illegible]