



Structure des ordinateurs et applications

À propos du cours

- **Crédits:** 02
- **Coefficients:** 02

Exercices (Vecteurs)

Exercice 1

- Traduire en langage c;derouler pour n=4;
v={3,5,9,7} ;
- Rréécrire avec while
et do.. while;
- Le programme calcule
combien d'éléments sont
divisible par 3;

Algorithme Vecteur;

Variables

V : Tableau [0..99] d'entier ;

N,i, C : Entier;

Début

{-*- Entrées -***-}**

Ecrire ('Spécifiez la taille de V: ');

Lire(N);

Ecrire ('Donner les éléments de V: ');

Pour i←0 à N-1 faire

Lire(V[i]);

FinPour;

{-*- Traitements -***-}**

C← 0;

Pour i←0 à N-1 faire

Si (V[i] mod 3 ≠ 0) alors

C← C+ 1;

FinSi; FinPour;

{-*- Sorties -***-}**

Ecrire (' C=', C) ;

Fin.

Exercice 2

Écrire un programme en C qui demande à l'utilisateur de saisir un entier N , puis lit N entiers et les stocke dans un tableau V. Le programme parcourt ensuite le tableau pour compter le nombre de valeurs positives, négatives et nulles, puis affiche ces résultats.

Exercice 3

Écrire un programme C qui lit un entier **N** représentant la taille d'un vecteur, puis lit **N** entiers et les stocke dans ce vecteur. Ensuite, le programme affiche les éléments du vecteur dans l'ordre inverse, du dernier au premier.

Exercice 4

Écrire un programme C qui lit un entier **N** représentant la taille d'un vecteur, puis lit **N** entiers et les stocke dans ce vecteur. Le programme doit ensuite calculer séparément la somme des éléments pairs et la somme des éléments impairs, puis afficher ces deux résultats.

Algorithme Tri_Selection

Variables

T : Tableau[0..99] de réel

I, J, N : entier

Z : réel

Début

Lire(N);

pour i ← 1 à N-1 faire

 Lire(T[i]);

Fin-Pour;

pour i ← 1 à N-2 faire

 pour j ← i+1 à N-1 faire

 Si T[i] > T[j] Alors

 Z ← T[i];

 T[i] ← T[j];

 T[j] ← Z;

 Fin-Si;

 Fin-Pour;

Fin-Pour;

pour i ← 1 à N faire

 Ecrire(T[i]);

Fin-Pour;

Traduire en programme en C

```
#include <stdio.h>
```

```
int main() {
```

```
float T[100];
```

```
int i, j, n;
```

```
float Z;
```

```
printf("Donner le nombre d'elements: ");
```

```
scanf("%d", &n);
```

```
printf("Donner les elements du tableau:\n");
```

```
for (i = 0; i < n; i++)
```

```
    scanf("%f", &T[i]);
```

```
for (i = 0; i < n - 1; i++) {
```

```
    for (j = i + 1; j < n; j++) {
```

```
        if (T[i] > T[j]) {
```

```
            Z = T[i];
```

```
            T[i] = T[j];
```

```
            T[j] = Z;
```

```
        }
```

```
    }
```

```
}
```

```
printf("Tableau trie:\n");
```

```
for (i = 0; i < n; i++)
```

```
    printf("%8.2f ", T[i]);
```

```
printf("\n");
```

```
return 0;}
```