

Exercice 1.

- Considérer le programme C suivant :

```
#include <stdio.h>
#include <stdlib.h>
#define TAILLE 100000000
char A[TAILLE];
int main (void)
{ for (int i=0; i<NSIZE; i++)
    A[i] = 'A';

    printf ("FAIT\n");
    return 0;
}
```

- Quelle serait la taille de l'exécutable correspondant en mémoire secondaire.
- Les 100000000 éléments du vecteur externe A sont-ils stockés dans l'exécutable ?

Exercice 2.

- Considérer le programme C suivant :

```
#include <stdio.h>
#define MAX 3
void somme(void);
int main()
{ int i;
  for(i = 0; i < MAX; i++)
    somme();
  return 0;
}

void somme(void)
{ static int total = 0;
  int nombre;
  printf("\n Entrer un nombre: ");
  scanf("%d", &nombre);
  total += nombre;
  printf("\n Le total est: %d\n", total);
}
```

- Si vous entrez successivement les 3 nombres suivants : 10, 20, 30 quel sera le résultat du programme précédent ? Pourquoi ?
- Enlever le mot-clé static de la première déclaration de la fonction somme. Ré-exécuter ce programme avec les mêmes données, quel sera le résultat ?

Exercice 3.

- Quelle sera la sortie du programme suivant ? Expliquez.

```
#include <stdio.h>
void fonction_1(void);
void fonction_2(void);
int globvar = 10;
int main()
{ globvar = 20;
  printf("\nVariable globvar, in main() = %d\n", globvar);
  fonction_1 ();
  printf("\nVariable globvar, in main() = %d\n", globvar);
  fonction_2 ();
  printf("\nVariable globvar, in main() = %d\n", globvar);
  return 0;
}

int globvar2 = 30;
void fonction_1 (void)
{ char globvar;
  globvar = 'A';
  globvar2 = 40;
  printf("\nIn fonction_1(), globvar = %c and globvar2 = %d\n", globvar, globvar2);
}

void fonction_2 (void)
{ double globvar2;
  globvar = 50;
  globvar2 = 1.234;
  printf("\nIn fonction_2(), globvar = %d and globvar2 = %.4f\n", globvar, globvar2);
}
```