

# EXERCICES SUR LES APPELS DE FONCTIONS ET LA PILE D'EXÉCUTION

## Exercice 1

### Question 1

Qu'est-ce qu'affiche le programme suivant ?

```
#include <stdio.h>

int diff(int a, int b) {
    int x = a - b;
    return x;
}

int calcul(int a, int b) {
    int x;

    if (a > b) {
        x = diff(a, b);
    }
    else {
        x = diff(b, a);
    }
    return x;
}

int main() {
    int res;

    res = calcul(7, 2);
    printf("Le premier resultat du calcul est %d.\n", res);
    res = calcul(-15, 3);
    printf("Le deuxieme resultat du calcul est %d.\n", res);
    return 0;
}
```

### Question 2

Donnez l'évolution de la pile d'exécution.

## Exercice 2

Donnez l'évolution de la pile pour le code suivant :

```
#include <stdio.h>

void ma_fonction(int a, int b) {
    int c1, c2;
    int *d;

    c1 = a + b;
    d = &a;
    *d = *d + 2;
    c2 = a + b;
    printf("a = %d, b = %d, c1 = %d, c2 = %d, *d = %d\n", a, b,
c1, c2, *d);
}

int main() {
    int a = 7, b = 10;

    printf("Avant appel : a=%d, b= %d\n", a, b);
    ma_fonction(a, b);
    printf("Après appel : a=%d, b= %d\n", a, b);

    return 0;
}
```

## Exercice 3

Donnez l'évolution de la pile pour le code suivant :

```
#include <stdio.h>

void ma_fonction(int *param1, int param2) {
    int var_loc = 3;

    *param1 = var_loc * param2;
    param2 = var_loc + 1;
}

int main() {
    int v1, v2;
```

```
v1 = 10;  
v2 = 3;  
ma_fonction(&v2, v1);  
  
return 0;
```

```
}
```