

CC - évaluation rapide (déc 2022)- **correction**

Énoncé J1 (en fait un des deux énoncés, le deuxième est similaire) N est une constante symbolique déclarée à l'avance.

1. Soit la fonction suivante :

```

int foo(int T[N], int m, int k){
    if ((k-m)<2) {return T[m];}
    else
        return T[m]+foo(T,m+1,k);
}

```

Quel calcul réalise cette fonction ? Vous prendrez au moins 2 exemples judicieusement choisis.

CORRECTION : Cette fonction calcule et renvoie la somme des éléments du tableau T situés entre les indices m et k compris. **en fait, en oubliant la case d'indice k , mais je n'ai pas compté faux en cas d'oubli de ce cas.** Par exemple, sur le tableau $t = [2, -1, 3, 4]$:

- l'appel `foo(t, 0, 2)` calcule $t[0] + \text{foo}(t, 1, 2)$, cad $t[0] + t[1]$, cad 1.
- l'appel `foo(1, 2)` calcule et renvoie $t[1]$, cad 2.

Paraphraser (ie répéter le code autrement le code n'est d'aucune utilité, et ne rapporte aucun point) ■

2. Écrire un programme principal qui déclare et initialise un tableau `tab` pour obtenir `[0,1,2,33,-1,8,10,-7]` puis appelle la fonction précédente, et affiche le résultat sur la sortie standard.

CORRECTION : **Attention à la taille du tableau.**

```

int main(){
    int tab[8]={0,1,2,33,-1,8,10,-7};
    printf("le resultat est=%d\n", foo(t,0,7));
    return 0;
}

```

■

Énoncé J2 (idem) N est une constante symbolique déclarée à l'avance.

1. Soit la fonction suivante :

```
bool bar(int T[N], int u){
    if (u==0) return (T[u]>0);
    else
        return (T[u]>0 && bar(T,u-1));
}
```

pour cette question considérons $N=3$. Que vaut la valeur booléenne $t[2] > 0$ lorsque le tableau t vaut $[9, 1, 2]$? Réaliser les étapes de calcul lors de l'appel `bar(t,2)` sur le même tableau . Que calcule la fonction dans le cas général ?

CORRECTION : La valeur vaut V (true) car $t[2]$ vaut 2. Les étapes de calcul (flèche = appel)

```
bar(t,2) --> true && bar(t,1) --> true && (true && bar(t,0))
               --> true && (true && true)
               renvoie true
```

Dans le cas général, la fonction calcule et renvoie “true” si et seulement si toutes les valeurs du tableau sont strictement positives. ■

2. On considère $N=6$. Écrire un programme principal qui déclare et initialise un tableau `tt` pour obtenir $[0, 1, 2, 33, -1, 8]$ puis appelle la fonction précédente et affiche le résultat (bool=entier) sur la sortie standard.

CORRECTION : **Attention à la taille du tableau.**

```
int main(){
    int tab[6]={0,1,2,33,-1,8};
    printf("le resultat est=%d\n", foo(t,0,5));
    return 0;
}
```

■