

Identification

Prénom:
Nom:
Code permanent:

Veillez choisir **une seule réponse**, avec une croix (×) ou un crochet (✓).

Q1. Considérez le programme suivant

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

int compare_int(const void* a, const void* b) {
    return *(int*)a - *(int*)b;
}

int main(void) {
    int t[] = {2, 3, 5, 7, 11};
    int a = 4;
    int* i = bsearch(/* TODO 1 */,
                    /* TODO 2 */,
                    5,
                    /* TODO 3 */,
                    compare_int);
    printf("%s\n", i == NULL ? "trouvé!"
           : "pas trouvé!");
}
```

Que doit-on mettre à la place de /* TODO 1 */ si on souhaite rechercher la clé 4?

- *a 1 &a a

Q2. Toujours avec le même programme, que doit-on mettre à la place de /* TODO 2 */?

- &t *t t &&t

Q3. Toujours avec le même programme, que doit-on mettre à la place de /* TODO 3 */?

- sizeof(int&) sizeof(int)
 sizeof sizeof(int*)

Q4. Que représente le type FILE introduit par la bibliothèque stdio?

- Un canal standard
 Un fichier en lecture/écriture
 Un fichier en lecture seule
 Un flux

Q5. Parmi les structures de données suivantes, laquelle correspond à un type abstrait?

- Une liste doublement chaînée
 Un ensemble
 Un tableau dynamique
 Un tableau statique

Q6. Quel nom donne-t-on à la propriété qu'une structure de données doit satisfaire avant et après qu'on lui ait appliqué une opération?

- Assertion Prédicat
 Invariant Spécification

Q7. Que se passe-t-il si on exécute le programme suivant?

```
#include <stdlib.h>

int main(void) {
    int* p = NULL;
    free(p);
    return 0;
}
```

- Le comportement est indéterminé
 Le comportement est normal
 Une boucle infinie
 Une erreur de segmentation

Q8. Que rapportera Valgrind sur la sortie standard si on l'exécute avec le programme suivant?

```
#include <stdlib.h>

int main(void) {
    int* t[2];
    t[0] = malloc(sizeof(int));
    t[1] = malloc(sizeof(int));
    t[0] = malloc(sizeof(int));
    free(t[0]);
    free(t[1]);
    return 0;
}
```

- Qu'il n'y a aucun problème de mémoire
 Qu'il y a une erreur de segmentation
 Qu'il y a une fuite de mémoire
 Qu'il y a une lecture interdite