

1 Pointeurs

Soit l'état de la mémoire suivant:

	360	340
	356	1
	352	336
	348	4
g	y[0]	4
	340	356
u	336	260
	332	1
	328	4
		...
	260	360

La première colonne donne le nom des variables, la deuxième donne des adresses mémoires (nous les exprimerons dans ce challenge en **base 10**) et, enfin, la troisième donne la valeur stockée à cette adresse. De plus, voici comment ont été déclarées les variables (on suppose que les entiers (`int`) sont représentés sur 4 octets¹):

```
1 int u, g, *ptr = &u, *q = &g;  
2 int y[] = {4,4,336,1};  
3 char *S = "informatique", *A = S + 3;
```

Dans ce Challenge, vous devez indiquer la valeur des **expressions** suivantes. Considérez qu'entre chaque expression, la mémoire est réinitialisée telle que présentée dans le schéma. Si un accès mémoire est demandé à une adresse hors de l'intervalle [260, 360], mentionnez l'erreur de segmentation par la valeur **SF**². Les adresses sont aussi représentées sur 4 octets.

¹les `short` sur 2, les `long` sur 8

²Pour Segmentation Fault.

1. `&g`
2. `y`
3. `&*q`
4. `* (int *)u`
5. `&* (int *)*ptr`
6. `*** (int ***)u`
7. `*** (int ***)*ptr`
8. `+++&g`
9. `--*ptr`
10. `(*ptr)++`
11. `*++q`
12. `(int*)u - g`
13. `ptr - &g`
14. `(int*)u - ptr[g]`
15. `(short)g + u`
16. `y[g]`
17. `*S`
18. `S[S[10] - A[-2]]`
19. `*(&g - g)`
20. `*(q - g)`