

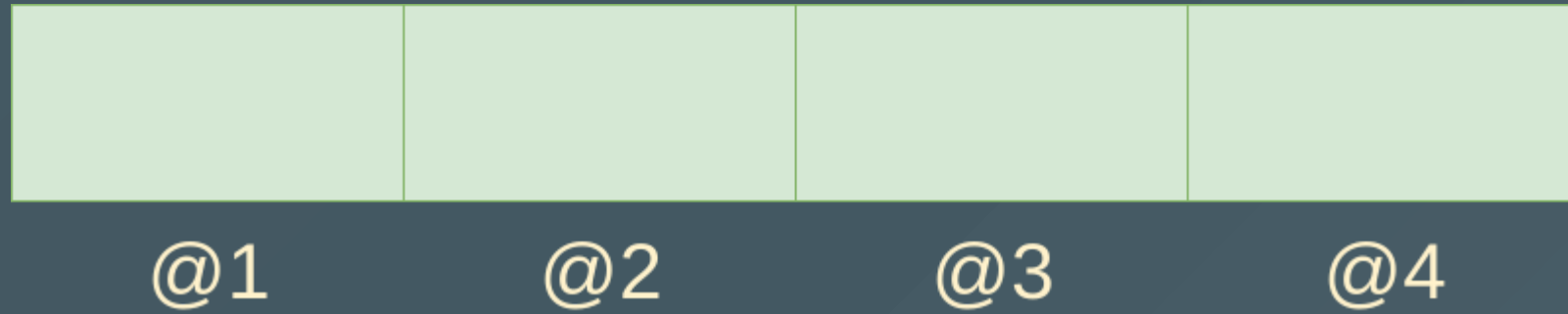
Le Langage Go

**Struct**

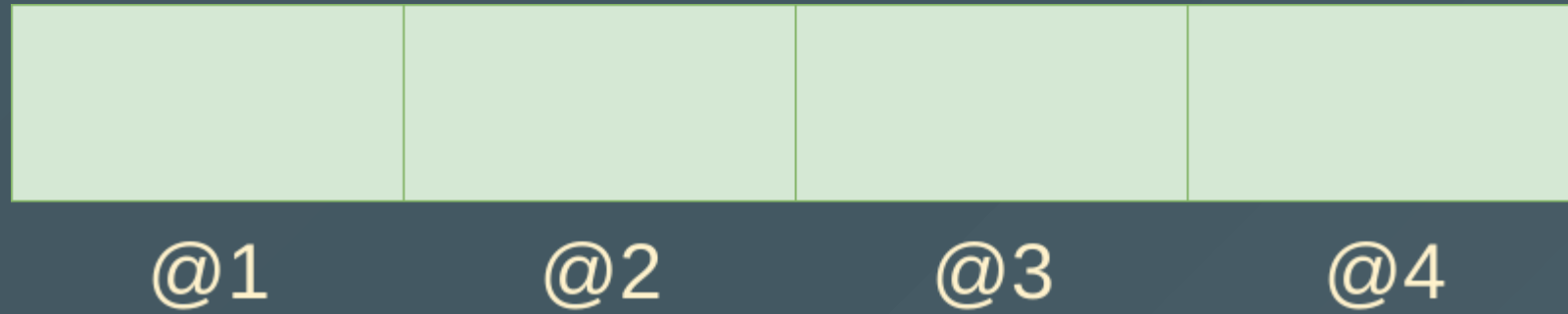
**Pointeurs**

# Comment fonctionne un pointeur ?

# Mémoire

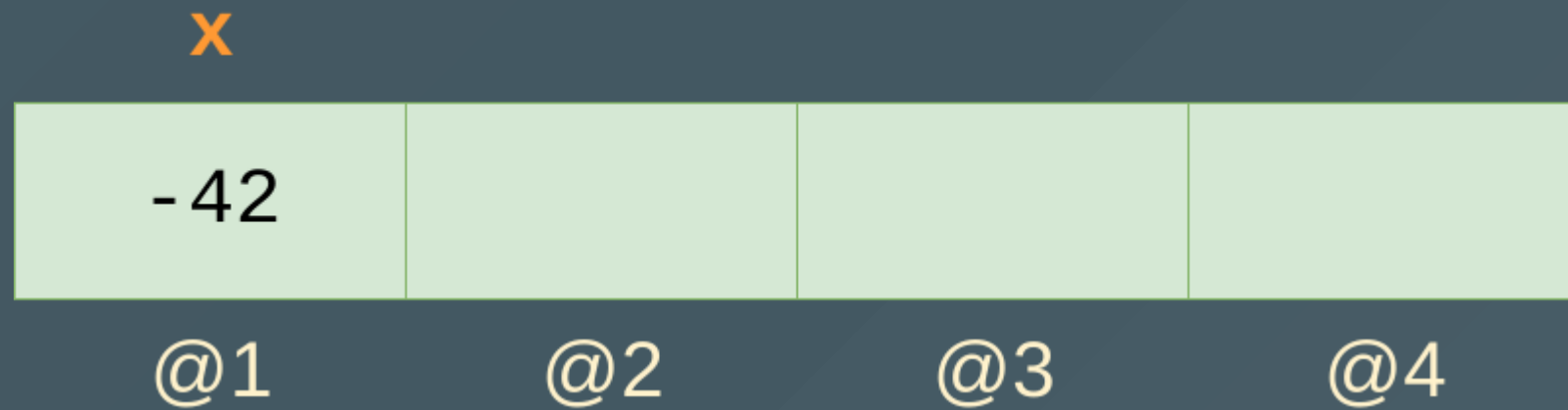


# Mémoire



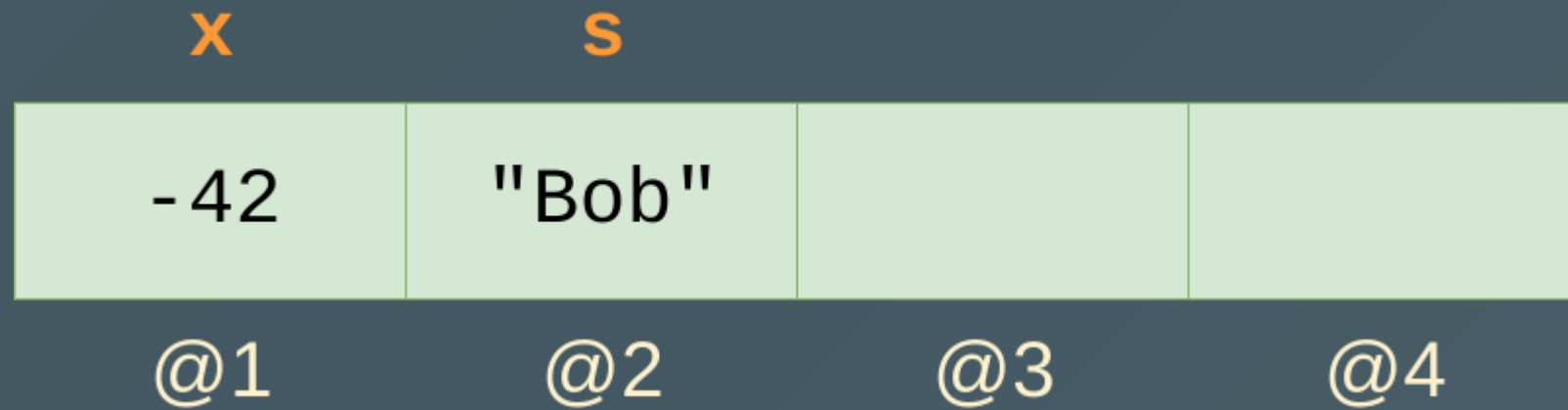
```
x := -42
```

# Mémoire



```
x := -42
```

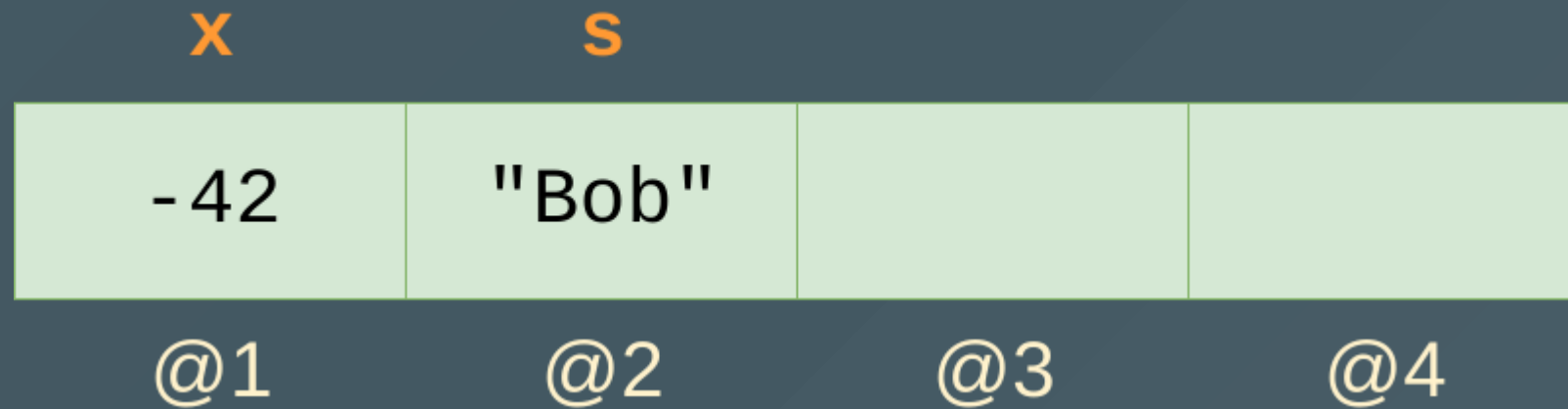
# Mémoire



```
x := -42  
s := "Bob"
```

**Un pointeur est**  
**une variable qui référence une adresse**

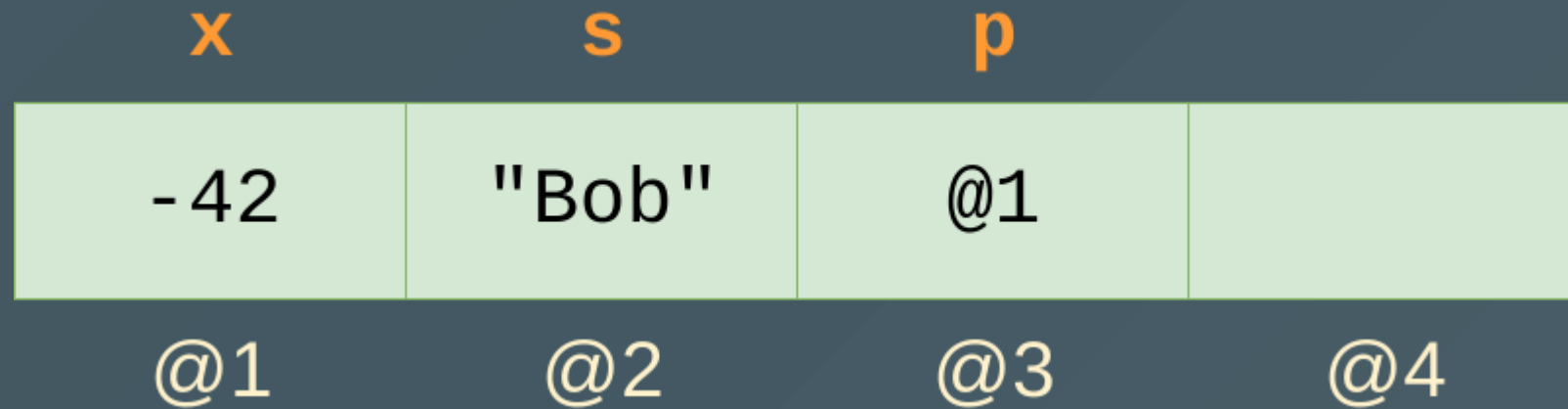
# Mémoire



```
x := -42
s := "Bob"
p := &x    // Création d'un pointer vers la variable x
```

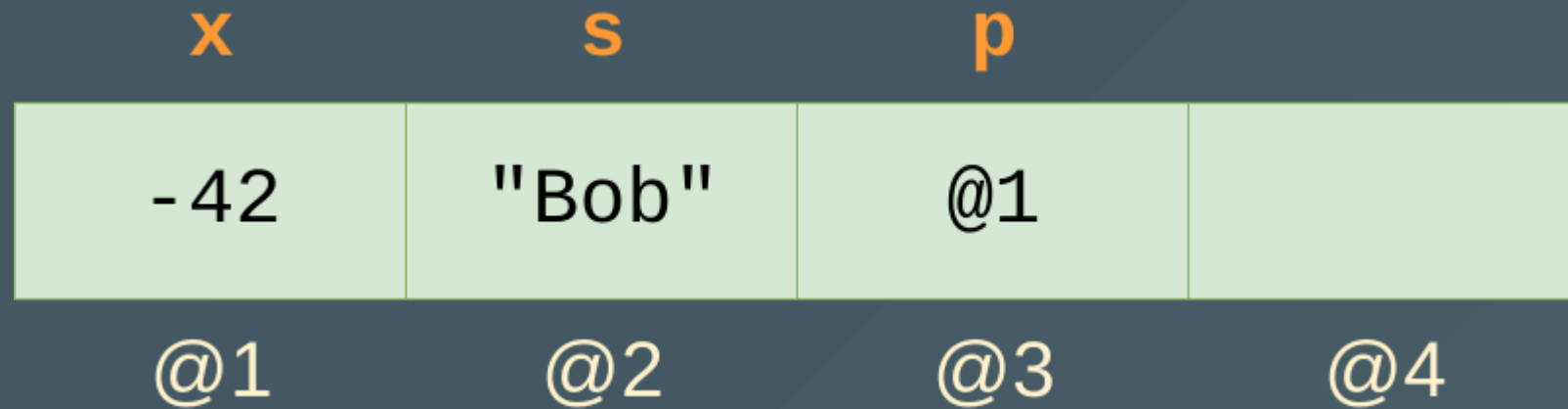


# Mémoire



```
x := -42
s := "Bob"
p := &x    // Création d'un pointer vers la variable x
```

# Mémoire



```
x := -42
s := "Bob"
p := &x    // Création d'un pointer vers la variable x
i := *p    // Déréférencement de p pour récupérer la valeur de x
```

# Mémoire

x	s	p	i
-42	"Bob"	@1	-42
@1	@2	@3	@4

```
x := -42
s := "Bob"
p := &x    // Création d'un pointer vers la variable x
i := *p    // Déréférencement de p pour récupérer la valeur de x
```

# Pourquoi s'embêter avec des pointeurs ?

**C'est le mécanisme royal  
pour partager  
et modifier la mémoire**

**Ils nous permettent de modifier des paramètres dans des fonctions**

**C'est exactement ce qu'on aimerait  
faire avec les méthodes**

