

# Référence du fichier exemple.cpp

---

Aller au code source de ce fichier.

## Fonctions

---

`int64_t ajouter (int64_t operandeA, int64_t operandeB)`  
Fonction d'addition de deux nombre de type `int64_t`. Plus de détails...

---

`int32_t ajouter (int32_t operandeA, int32_t operandeB)`  
Fonction d'addition de deux nombre de type `int32_t`. Plus de détails...

---

## Documentation des fonctions

---

```
int64_t ajouter ( int64_t operandeA,  
                int64_t operandeB  
                )
```

Fonction d'addition de deux nombre de type `int64_t`.

La fonction d'addition est polymorphe, c'est à dire qu'elle dépend du type des paramètres fournis. Celle-ci correspond au type `int64_t`.

### Paramètres

**operandeA** un entier 64bits

**operandeB** un entier 64bits

### Renvoie

La somme des deux entiers 64bits fournis en paramètres

### Voir également

**ajouter(int32\_t, int32\_t)**, **ajouter(double, double)**, **ajouter(float, float)**

```
13 | {  
14 |     return operandeA + operandeB;  
15 | }
```

```
int32_t ajouter ( int32_t operandeA,  
                  int32_t operandeB  
                  )
```

Fonction d'addition de deux nombre de type int32\_t.

La fonction d'addition est polymorphe, c'est à dire qu'elle dépend du type des paramètres fournis. Celle-ci correspond au type int32\_t.

### Paramètres

**operandeA** un entier 32bits

**operandeB** un entier 32bits

### Renvoie

La somme des deux entiers 64bits fournis en paramètres

### Voir également

**ajouter(int64\_t, int64\_t)**, ajouter(double, double), ajouter(float, float)

```
29 | {  
30 |   return operandeA + operandeB;  
31 | }
```