

Norme Indie Project (Suck My Wheels)

Sommaire :

1. Case
2. Namespace
3. Commentaires
4. Includes
- 5. MENACES ET REGLEMENTS DE COMPTES**
6. Antipaki

NORME CASE

```
void ceci_est_une_fonction();
void ceciEstUneMethode();
int _ceciEstUnAttributPrivate;
int moiJeSuisPublic;
class JeSuisUneClass
{
};
```

```
struct je_suis_une_struct_t (UNDERSCORE T !)
```

```
{
}
```

```
enum je_suis_le_vomi_type (UNDERSCORE TYPE !)
```

```
{
    CAPSLOCK,
    BITE
};
```

```
#define CAPSLOCK bite
#define MACRO(A, B) ((A) + (B))
```

Fichier d'une classe : NomDeLaClass.hpp

NORME NAMESPACE

Fonction seules -> Namespace core (si une fonction est dans un namespace, pour savoir de quel .hpp il vient)

Pas de using (sauf pour l'héritage)

NORME COMMENTAIRES

Fonction ou namespace dans les .hpp :

```
/* bonjour je suis une fonction  
   et je sers à bite */
```

Commentaire suite à une condition (pour les grosses conditions) :

```
if (mes_couilles) //bonjour, je fais tel et tel truc  
{  
}
```

```
while (ma_bite < 42) //tant que j'ai pas 42km de queue  
{  
}
```

NORME INCLUDES

Lib CPP :

```
#include <bite>
#include <cul>
```

Lib C :

```
extern "C"
{
    #include <bite.h>
    #include <cul.h>
}
```

Lib perso :

```
#include "bite.hpp"
#include "cul.hpp"
```

Prototypes :

```
#include "bite.hpp"
#include "cul.hpp"
```

Si on a besoin d'utiliser une variable ou un type d'un .hpp quelconque dans un autre .hpp, NE PAS INCLUDE. On ne prototype pas les fonctions et les class templatées. Plus simplement, faites :

class JeSuisUneClasse;

NORME MENACES & REGLEMENT DE COMPTES

(par mas_b en personne)

Les (?:) et les (,) dont les retours ne sont pas utilisés seront sévèrement punis. Genre, vraiment.
Un ternaire pour économiser des lignes ou faire des appels de fonctions dans le ternaire et ne pas utiliser leurs valeurs de retours...

Créez des defines pour les valeurs en dur !!!! return(42) ; ne sera pas toléré.

this-> et NULL sont obligatoires !

[MAS_B INTENSIFIES]

<https://media.giphy.com/media/xT1XGXM1JNF1C79xXa/giphy.gif>



NORME ANTIPAKI (nous devons retrouver des espaces mémoires nationaux)

buff size -> sizeof(type) * (size) ;
union -> char[sizeof(type)];

BONNE BITE A TOUS !

اجم يع في طيبة BITE

ALL 좋은 BITE!