



Préambule

Pour vous permettre de mieux retrouver vos fichiers tout au cours de l'année, je vous conseille vivement de créer un répertoire, par exemple ProgC , et dans ce répertoire de créer un sous répertoire par TP.

Premier programme

- ① Dans l'éditeur de votre choix (*xemacs*, *gedit*, *kate*, *vi* ...), créez le programme qui correspond à l'algorithme 1. Une fois ceci fait, compilez le programme obtenu. Enfin, exécutez le programme obtenu. ☐

Algorithme 1 Hello World!

```
programme Bonjour  
    écrire ("Bonjour tout le monde")  
fin programme
```

- ② Il y a de forte probabilité que le programme que vous venez d'écrire ne respecte pas les normes de codage imposées à l'EISTI. Récupérez ces normes, lisez les attentivement, et corrigez votre programme en conséquence. ☐

Séquences d'échappement

- ③ Déterminez à quoi servent les séquences d'échappement exposées en cours. ☐

Remarques:

1. Pour compiler, on utilise la commande `gcc` avec les options suivantes :

-Wall : affiche tous les warnings de construction

-o *file* : spécifie le fichier *file* de sortie, qui correspond au programme à exécuter

```
gcc -Wall nomFichier.c -o nomProgramme
```

Pour exécuter, on lancera le programme de la façon suivante :

```
./nomProgramme
```

2. La génération de la documentation doxygen nécessite un fichier de configuration, généralement nommé *Doxyfile* ou *doxyfile*.

— si le fichier de configuration *Doxyfile* n'existe pas, le générer

```
doxygen -g
```

Après avoir adapté le fichier de configuration au cas d'utilisation (champs `PROJECT_NAME`, `PROJECT_NUMBER`, `OUTPUT_DIRECTORY`, ...), compilez la documentation.

```
doxygen Doxyfile
```

Lors de la compilation, il peut y avoir des warnings ou des erreurs de compilation. Vérifiez que tout s'est bien déroulé. En fonction de ce que vous avez demandé de générer comme documentation, la trouverez soit en format \LaTeX , soit en format HTML, ...