

TP 1 — Premiers pas en C++

I Utilisation de l'IDE geany pour le C++

Geany est un IDE léger vous permettant de développer en langage C/C++. Geany est pré-installé sur vos machines de TP. Pour l'utiliser :

1. Ouvrir geany via le finder ou via la commande

```
sysadmin@localhost:~$ geany&
```

2. **À faire uniquement la première fois.** Allez dans Éditer → Préférences → Outils et changez la ligne Terminal par celle-ci :
`x-terminal-emulator -e /bin/sh %c`
3. Allez dans Fichier → Nouveau (selon un modèle) → main.cxx
4. Vous pourrez ensuite taper votre code dans la fonction main
5. La touche F9 sert à compiler et la touche F5 à exécuter le code.
Attention! Pensez à compiler avant d'exécuter!

II Exercices : éléments de base du langage

Exercice 0 : Écrire un programme TP1 qui affiche ceci à l'écran :

```
sysadmin@localhost:~$ ./TP1
Hello world!
Voici un programme illustrant l'utilisation de std::cout!
```

Exercice 1 : Écrire un programme qui demande à l'utilisateur de saisir la largeur et la longueur d'un rectangle et qui en affiche le périmètre et la surface. La largeur et la longueur seront entiers.

Exercice 2 : Écrire un programme qui demande à l'utilisateur de saisir deux entiers A et B, puis qui échange les contenus des variables A et B, et enfin qui affiche A et B.

Exercice 3 : Écrire un programme qui demande à l'utilisateur de saisir un entier et qui affiche GAGNÉ si l'entier est entre 56 et 78 (bornes incluses), et qui affiche PERDU sinon.

Exercice 4 : Écrire un programme qui demande à l'utilisateur de saisir 5 entiers et qui affiche leur moyenne. Essayer d'utiliser le moins de variables possible. Essayer en n'utilisant que 2 variables.

Exercice 5 : Écrire un programme qui affiche tous les entiers de 8 jusqu'à 23 (bornes incluses) en utilisant une boucle `for`, puis qui affiche uniquement les nombres pairs avec une deuxième boucle `for`. Obtenir de nouveau ce résultat avec deux boucles `while`.

Exercice 6 : Écrire un programme qui demande à l'utilisateur de saisir 5 entiers inférieurs à 200, puis qui affiche le plus petit de ces entiers.

Exercice 7 : Écrire un programme qui demande à l'utilisateur de saisir 5 entiers stockés dans un tableau. Le programme doit afficher le nombre d'entiers supérieurs ou égaux à 10.

Exercice 8 : Écrire un programme qui demande à l'utilisateur de saisir 5 entiers, qui seront stockés dans un tableau, ainsi qu'un entier V. Le programme doit rechercher si V se trouve dans le tableau, puis afficher sur la sortie standard "V se trouve dans le tableau" ou "V ne se trouve pas dans le tableau".

Exercice 9 : Écrire un programme qui demande à l'utilisateur de saisir 5 entiers, qui seront stockés dans un tableau. Le programme doit ensuite afficher le plus grand élément du tableau ainsi que son indice.

Exercice 10 : Écrire un programme qui demande à l'utilisateur de saisir 5 entiers puis affiche le nombre d'occurrences du plus grand d'entre eux.

Exercice 11 – Pointeurs : Soient trois entiers a, b et c et deux pointeurs de type entier p_1 et p_2.

— Déterminez quelle est la valeur de chacune des opérations listées dans le tableau ci-dessous. Les opérations sont à faire à la suite et dans l'ordre indiqué.

Opérations	a	b	c	*p_1	*p_2
Initialisation	1	2	3	?	?
p_1 = &a					
p_2 = &c					
*p_1 = *p_2					
(*p_2)++					
p_1 = p_2					
p_2 = &b					
*p_2 = *p_1 - 2 * *p_2					
(*p_2)--					
*p_1 = *p_2 - c					
a = (2 + *p_2) * *p_1					
p_2 = &c					
*p_2 = *p_1 / *p_2					
*p_1 = a + b					
a += *p_1					
b = *p_1 + *p_2					
*p_1 = 2 * a					
a = *p_2					
*p_1 = 1 - c					
*p_2 += *p_1 + a					

— Vérifiez que vos réponses sont correctes en écrivant le code permettant de les calculer.

Exercice 12 : Écrire un programme qui permet à l'utilisateur de gérer une liste d'entiers. La liste, un tableau de 10 entiers, est initialisée à 0. Ensuite, le programme propose les options suivantes :

1. Ajouter un entier dans la liste¹
2. Afficher la liste
3. Afficher la moyenne des éléments de la liste
4. Supprimer tous les entiers ayant une valeur donnée²
5. Quitter

1. par exemple, ajouter 5 à {1 2 3 0 0 0 0 0 0 0} donne {1 2 3 5 0 0 0 0 0 0}

2. par exemple, supprimer les occurrences de 5 dans {13 5 37 5 -23 42 -5 1 5 7} donne {13 37 -23 42 -5 1 7 0 0 0}