

C++, Feuille de TP n° 2 : Conditionnelles

Objectifs

1. Tester ses programmes
2. Raisonner avec des conditionnelles
3. Respecter la syntaxe et les règles de mise en forme

Consignes générales pour les exercices sur machine

1. Lisez bien l'énoncé avant de vous mettre à coder
2. Chaque question doit être testée avant de passer à la suivante avec un jeu d'essais le plus complet possible.
3. Ajouter en commentaire en fin de programme, la liste des différents tests effectués, avec une explication du choix de chaque test (une ligne par test).
4. Vous référer au poly C++ Volume 1 pour les détails de la syntaxe

TRAVAIL à RENDRE : l'exercice 4, code source et jeu d'essais complet

Exercice 1 : De l'ordre

Ecrire un programme qui demande à l'utilisateur de saisir deux valeurs réelles **a** et **b** puis qui affiche sur le flot de sortie « ordre croissant » si **a** est plus petit ou égal à **b**, et « ordre décroissant » si **b** est plus petit que **a**. Bien tester tous les cas de figure (exemples : 1 et 5.5 et 1.5 et 5), et les noter en commentaire en fin de fichier.

Exercice 2 : Beau temps pour la saison

On souhaite un programme qui demande de saisir une température et affiche "Il fait trop chaud :-(" si celle-ci est supérieure à 25, "Il fait trop froid :-(", si celle-ci est inférieure à 15, et "Cette température me convient :)" si elle est comprise entre 15 et 25. Commencer par préparer le jeu d'essais puis écrire le programme en recopiant les essais en commentaire en fin de fichier.

Exercice 3 : Extremums parmi 3

On souhaite écrire un programme qui demande de saisir trois nombres **x**, **y** et **z** et affiche la plus grande et la plus petite des valeurs. Utiliser les variables indiquées ci-dessous.

- a) Écrire le jeu d'essais
- b) Écrire le programme pour afficher le max. Vérifier avec le jeu d'essais

- c) Compléter pour afficher aussi le min. Vérifier avec le jeu d'essais.

```
int main() {
    double mini, maxi;
    double x, y, z;
    ...
}
```

Exercice 4 : Quel spectacle ! (à rendre)

Déposer sur Moodle le programme commenté accompagné du tableau du jeu d'essais (en commentaire à la fin du programme)

Une salle de spectacles propose la grille tarifaire ci-dessous qui dépend de l'âge de l'individu, du fait qu'il soit abonné ou pas, et du type de spectacle (court ou long : 'c' ou 'l'). Écrire un programme qui, après avoir demandé au client le type de spectacle, l'âge du spectateur et s'il est abonné ou pas, affiche le prix du billet. Pour pouvoir gérer facilement les évolutions de tarifs, les différents tarifs applicables seront mémorisés dans des constantes. Il est conseillé d'utiliser une variable prix.

NB : Il est demandé d'utiliser des conditionnelles imbriquées dès que cela est possible pour éviter tester 2 fois une condition déjà traitée. Pour cela, ne tester qu'une seule condition à la fois.

- Écrire la partie saisie du type de spectacle, l'âge du spectateur, et s'il est abonné ou pas. Pour le type de spectacle déclarer une variable `typeSpectacle` de type `char`. Indiquer à l'utilisateur qu'il doit saisir 'c' ou 'l'. Pas de vérification, mais passer en minuscules (`tolower(typeSpectacle)` avant de tester. Pour abonné utiliser de même une variable de type `char` et de nom `abonne` ('o' ou 'n').
- Compléter pour calculer et afficher le tarif, sans tenir compte du type de spectacle. Tester les différents cas possibles.
- Compléter pour prendre en compte le type de spectacle en utilisant un `switch`. Tester l'ensemble des cas.

	Spectacle court		Spectacle long	
	abonné	non abonné	abonné	non abonné
Enfant (<18 ans)	6	8	10	12
Adulte	15	18	18	22

Exercice 5 : (optionnel, pour s'exercer) les saisons

En utilisant l'instruction `switch`, écrire un programme qui lit un caractère ('p' pour printemps, 'e' pour été, 'a' pour automne, 'h' pour hiver) et affiche le nom complet de la saison correspondante. Le programme affichera sur le flot de sortie un message d'erreur si la lettre tapée ne correspond pas à une saison.

Exercice 6 : (optionnel) pour les plus rapides Ordinaux abrégés en anglais

Dans cet exercice, nous nous intéressons aux ordinaux abrégés, où le nombre est écrit en chiffres.

Les premiers sont « 1st », « 2nd », « 3rd », « 4th », etc. (abréviations de « first, second, third, fourth, ... »).

Pour déterminer le suffixe, on regarde le dernier chiffre du nombre : si c'est 1, on ajoute le suffixe est -st ; si c'est 2, le suffixe est -nd ; si c'est 3, le suffixe est -rd ; sinon le suffixe est -th.

Il y a une exception : si l'avant-dernier chiffre du nombre est 1, le suffixe est toujours -th.

Écrire un programme qui lit un nombre, et qui affiche l'ordinal anglais abrégé correspondant.

Indication : le dernier chiffre d'un nombre `val` vaut exactement `val%10` (`val%n` – `val` modulo `n` – est le reste de la division entière de `val` par `n`).