

SEANCE 1 - DECOUVERTE DE LA STRUCTURE IF

OBJECTIFS

- Exprimer une condition, conditionner du code
- Indenter du code
- Apprendre à tester correctement son code
- Être critique de son propre code en vue de critères de qualité

EXO 1 : LECTURE DE CODE

1. D'après la lecture du code ci-dessous, **complétez** le jeu d'essai en renseignant la colonne résultat :

Données en entrée		Résultat
X	Y	
4	4	2
4	3	1
5	2	8
5	1	6
10	1	12

```
// x,y : noms pas explicites car ce code n'a pas de sens
// si ce n'est de tester votre logique d'exécution
int x =4,y =4 ,res=0;
if (x<5)
{
    if (y>3)
        res=1;
    if (x-y>0)
        res=0;
    res ++ ;
}
else if (x<10)
{
    res =x+2;
    if (y>=2)
        res ++;
    else
        res --;
}
else
    res =12 ;
Console.WriteLine("res : " + res);
```

2. Au sein du répertoire « **R1_01_InitiationDev_C#** », créez un répertoire « **Sequence_2_StructuresConditionnelles** » et puis créez un projet « **Exo1** » dans une solution « **Seance1_Decouverte** », copiez collez le code, changez le valeurs et vérifiez. Si vous n'obtenez pas le bon résultat, faites du pas à pas (mode debugge : F11) pour suivre l'évolution des variables.

EXO 2 : PROGRAMME DE SOLDES

Ajoutez un projet « **Exo2_Soldes** » à votre solution, celui-ci doit calculer des remises et prix finals pendant des périodes de soldes. Une boutique décide pendant une semaine de brader **tous** ses articles : les articles étiquetés rouges seront à -50 % tandis que les articles étiquetés verts seront à - 20 %. Votre programme doit afficher la remise et le prix final de l'article acheté.

Remarques :

- les taux de remise doivent être considérés comme des constantes.
- Votre affichage du résultat doit être sorti de tout bloc d'instructions conditionné { }
- On ne gère pas les erreurs de saisie ! (il n'y a pas d'erreur)

Puis vérifiez avec le jeu de tests suivant :

Données en entrée		Résultats attendus	
couleur	prix	remise	prixFinal
R	99	49,50	49,50
V	80	16	64

EXO 3 : STATUT

Ajoutez un projet « **Exo3_Statut** » à votre solution, Votre programme doit afficher en fonction de l'âge de l'utilisateur de type **uint**, le statut de la personne : enfant (de 0 à 9 ans), pré-adolescent(de 10 ans à 13 ans), adolescent(de 14 à 17 ans) ou bien adulte. Essayez de trouver une solution qui minimise le nombre de test.

Puis vérifiez avec le jeu de tests suivant : Remarque : quand il y a des intervalles, il faut tester les valeurs médianes mais aussi les valeurs aux bornes

Données en entrée	Résultats attendus
age	Affichage
5	Enfant
9	Enfant
10	Pré-ado
13	Pré-ado
14	Ado
17	Ado
18	Adulte

EXO 4 : MINIMUM

Ajoutez un projet « **Exo4_Min** » à votre solution. Votre programme doit afficher le plus petit de trois nombres entrés au clavier par l'utilisateur. Essayez de trouver une solution qui minimise le nombre de test.

LECTURE DE CODE

Que contient tmp ? Que font ces instructions ?

```
int a = 12, b = 18, c = 20 ;
tmp = a ;
if ( b >= tmp )
    tmp = b ;
if ( c >= tmp )
    tmp = c ;
```

EXO 5 : APPELS TELEPHONIQUES

Ajoutez un projet « **Exo5_AppelTelephonique** » Votre programme doit afficher le prix d'un appel téléphonique selon la grille tarifaire ci-dessous. Rem : On préférera une solution avec **imbrication**. On ne gère pas les erreurs de saisie ! (il n'y a pas d'erreur)

L'utilisateur doit saisir :

- La durée de son appel : demandez heure et minutes
- La destination de l'appel : international/national
- La tarification liée au moment de la journée : heure creuse / heure pleine

prix à la minute d'un appel	Heure creuse(C)	Heure pleine(P)
Appel national (N)	10 centimes	20 centimes
Appel international (I)	15 centimes	23 centimes

Données en entrée			Résultats attendus