

Nom, prénom : _____ Groupe : _____

TP 1 : Eléments de base

Préparation

Pour chacun des programmes suivants, trouver ce qui va s'afficher :

a.

```
void main()
{
    printf ("Mon premier programme !");    // Message
    getch();
}
```

> _____

b.

```
void main()
{
    char i=100,j;           // Déclaration des variables
    j=i+1;                  // J <- i+1
    printf ("%d+1=%d",i,j); // Affichage des variables
    getch();                // Attend une touche
}
```

> _____

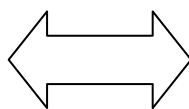
c.

```
void main()
{
    char ch='A';
    printf ("Le caractère %c a pour code ASCII %d",ch,ch);
    getch();
}
```

> _____

d.

```
void main()
{
    unsigned char a=7;
    a=~a;
    printf ("%d",a);
    getch();
}
```



a= ()₂

~a= ()₂= ()₁₀

> _____

e.

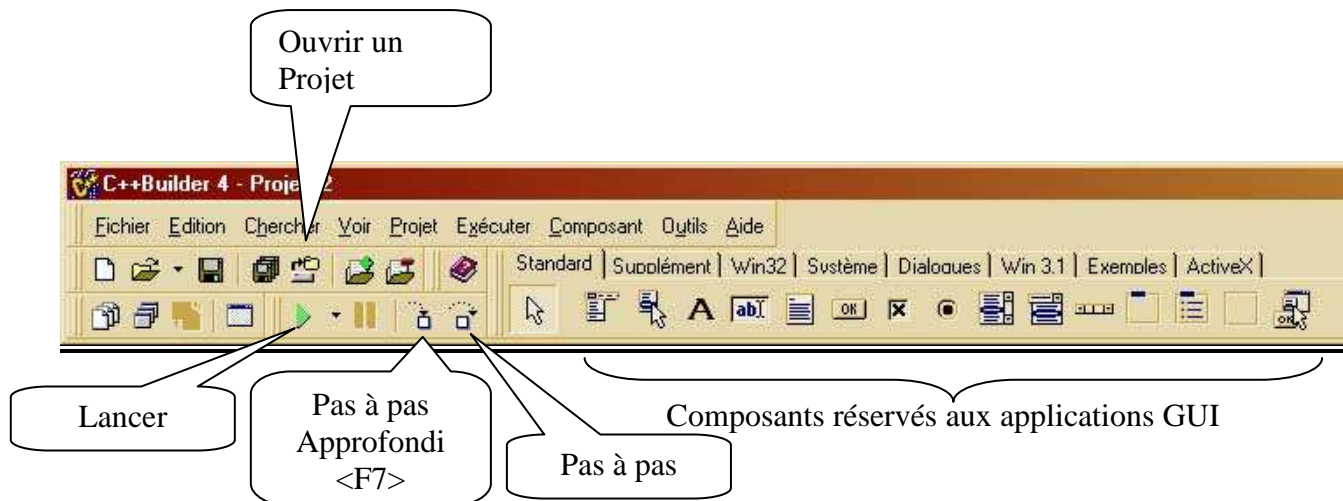
```
void main()
{
    unsigned char i=100,j,k;
    j=i*3;
    k=j/3;
    printf ("%d*3/3=%d",i,k);
    getch();
}
```

> _____

Travail pratique

C++Builder est un outil de développement intégré, il permet de construire des applications capables de s'exécuter sous l'environnement Windows. Il existe, pour ce qui nous concerne, deux types d'applications :

- Application Console s'exécutant dans une fenêtre noire type DOS
- Application GUI (Graphic User Interface) Application graphique Windows



Un projet est une sorte de boîte qui contient les divers fichiers qui doivent servir à construire votre application. Le fichier source que vous rédigez n'est qu'une partie du contenu du projet. Voici les principaux menus de Builder 4 :

Menu **Fichier** :

Nouveau : permet de créer un nouveau projet. Ne pas choisir *Nouvelle application* qui crée un nouveau projet graphique. Dans le cadre des TP, choisissez *Expert Console*, choisir *Application Console* et *.EXE*.

Ouvrir un Projet : Lorsque vous voulez continuer un travail déjà commencé.

Divers Enregistrer : Pour enregistrer votre travail, fichiers sources, projets etc...

Menu **Edition** :

Copier : <Ctrl+C>

Coller : <Ctrl+ V>

Couper : <Ctrl+X> Préférer cette commande au <Suppr>, si vous changez d'avis...

Menu **Voir** :

Gestionnaire de Projet : Utile pour gérer les fichiers qui appartiennent à votre projet.

Menu **Projet** :

Compiler l'unité : Traduit le fichier source ouvert après en avoir vérifié la syntaxe.

Ajouter au projet : Permet entre-autres d'insérer les pilotes d'E/S dans votre projet.

Make pour le projet : <Ctrl+F9> Relie les divers fichiers du projet.

Construire le projet : Idem, mais reprend tout le travail à zéro.

Menu **Exécuter** :

Exécuter <F4> Lance l'exécution du programme à vitesse normale.

Pas à Pas <F8> Avance de ligne en ligne.

Pas à Pas approfondi <F7> Permet de pénétrer à l'intérieur de fonctions à vérifier à condition que leur code soit disponible.

Aller jusqu'au curseur <F4> Exécute le code jusqu'à la ligne où se trouve le curseur.

Suspendre l'exécution.

Réinitialiser le Programme <Ctrl+F2> Quand votre programme s'est terminé de façon anormale.

Ajouter un point de suivi <Ctrl+F5> Important...Permet de visualiser le contenu des variables du programme.

Exercice 1 : Lancer l'exécution de CBuilder au moyen de Windows ou du raccourci sur le bureau. Dans le menu Fichiers, sélectionner Nouveau puis Application Console et .exe. Valider. Une fenêtre s'ouvre avec le contenu ci-dessous.

```
#pragma hdrstop
#include <condefs.h>
#pragma argsused

int main(int argc, char* argv[])
{
    return 0;
}
```

Remplacer le contenu généré automatiquement par celui-ci :

```
#include <stdio.h>                                // Librairie nécessaire pour printf
#include <A COMPLETER>                             // Librairie nécessaire pour getch

/* Programme principal */

void main()
{
    printf ("Mon premier programme !");           // Message
    getch();
}
```

Pour fonctionner correctement, le compilateur a besoin de savoir dans quelle librairie se trouve les fonctions printf() et getch(). Placer le curseur sur la fonction printf et appuyer sur F1, l'aide de cette fonction apparaît. En procédant de la même façon pour getch, déterminer de quelle librairie le programme a besoin pour fonctionner.

Sauvegarder vos projets sous les noms TPX_ExoY.bpr dans votre répertoire de travail, X étant le numéro du TP, Y le numéro de l'exercice. Vérifier ensuite que la sauvegarde contient bien le projet et le code source.

Exercice 2 : Créer un nouveau projet, saisir et exécuter le programme suivant après avoir ajouté les inclusions (#include <...>) :

```
void main()
{
    char i=100,j;           // Déclaration des variables
    j=i+1;                 // j <- i+1
    printf ("%d+1=%d",i,j); // Affichage des variables
    getch();               // Attend une touche
}
```

Modifier le programme de façon à initialiser i à 127 au lieu de 100, pourquoi le résultat n'est pas cohérent ? Proposer une solution pour que le résultat soit correct.

Exercice 3 : Saisir et exécuter le programme suivant après avoir ajouté les inclusions :

```
void main()
{
    char ch='A';
    printf ("Le caractère %c a pour code ASCII %d",ch,ch);
    getch();
}
```

Modifier le programme de façon à déterminer les codes ASCII des caractères 0,1,2,c,C,@ et du point d'interrogation.

Modifier le programme de façon à pouvoir trouver les caractères correspondants aux codes suivants : 41, 176, 138, 120, 121, 122, 228.

Exercice 4 : Ecrire un programme qui permet de calculer la valeur de x (éliminer les parenthèses superflues):

$$x = \frac{125,5 + 12,5}{5468,12} - 12,5 * 2,54$$

Exercice 5 : Saisir et exécuter le programme suivant après avoir ajouté les inclusions :

```
void main()
{
    unsigned char a=7;
    a=~a;
    printf ("%d",a);
    getch();
}
```

Expliquer ce résultat.

Exercice 6 : Saisir et exécuter le programme suivant après avoir ajouté les inclusions :

```
void main()
{
    unsigned char i=100,j,k;
    j=i*3;
    k=j/3;
    printf ("%d*3/3=%d ??? ",i,k);
    getch();
}
```

Expliquer ce résultat.

Exercice 6 : Saisir et exécuter le programme suivant après avoir ajouté les inclusions :

```
void main()
{
    unsigned char i=100,j,k;
    j=i*3;
    k=j/3;
    printf ("%d*3/3=%d ??? ",i,k);
    getch();
}
```

Expliquer ce résultat.

Exercice 7 : Saisir et exécuter le programme suivant après avoir ajouté les inclusions :

```
void main()
{
    int i=5,j;
    j=++i;
    printf ("i=%d j=%d",i,j);
    getch();
}
```

Modifier le programme de façon à compléter le tableau fourni sur le document réponse.

Exercice supplémentaire : Saisir et exécuter le programme suivant après avoir ajouté les inclusions :

```
void main()
{
    int i=0;
    printf ("%d = %d?",++i,i++);
    getch();
}
```

Expliquer ce résultat.

Compte-rendu à compléter

Exercice 1 : Librairie nécessaire pour getch :

Exercice 2 :

Inclusions :

Explication :

Solution :

Exercice 3 :

Caractère	Code ASCII
0	
1	
2	
c	
C	
@	
?	

Code ASCII	Caractère
41	
176	
138	
120	
121	
122	
228	

Exercice 4 :

```
Programme : void main ()  
            {
```

```
            }
```

Résultat x=

Exercice 5 :

Explication :

Exercice 6 :

Explication :

Exercice 7 :

	Incrémentation	Décrémentation
Pré -	Instruction : $j=++i$; Affichage : _____	Instruction : _____ Affichage : $i=4 \ j=4$
Post -	Instruction : $j=i++$; Affichage : _____	Instruction : _____ Affichage : _____

Exercice supplémentaire :

Explication :