

Responsable
pédagogique

AF

AM

PB

PM

Période

**Sem
1**

Sem2

Sem3

Sem4

Volume horaire

Cours

TP

5

STRUF

TP second degré

Indicateur temporel (hors rédaction du compte-rendu) :

questions	1h	2h	3h	4-5h
1				
2				
3				

Documents à rendre : Compte-rendu contenant à minima les sources (avec entête standard) et les résultats obtenus.

Tous les fichiers de ce TP se trouvent dans le répertoire /pub/Modules/STRUF de la machine rimatara ; dans l'archive STRUF.zip

Etude du programme de résolution second degré

On analyse un programme de résolution d'une équation de second degré.

$$a \cdot x^2 + b \cdot x + c = 0.$$

L'utilisateur saisira les 3 coefficients entiers (ou réels) a,b,c, sur la ligne de commande.

Prendre les fonctions de argv.h IntegerArg(N) , ou FloatArg(N) pour saisir ces coefficients.

1.

2. On demande de coder en langage C la résolution dans le corps des réels.

Important : pour le calcul des racines ,il faut utiliser la racine carrée du delta , donc prendre la fonction `sqrt()`. de la librairie mathématique.

Faites un man sur `sqrt()`. Exemple `int b = sqrt(16) ; /* b vaudra 4 ! */`

La compilation se fera par `gcc -Wall -o 2degv1 2degv1.c -lm`

(Ne pas oublier `-lm` c'est un L minuscule, pas un 1 !!)

3. Faites évoluer votre code pour calculer les 2 racines si il y en a !. Pensez aux différents cas . Un lien : [Pour ceux qui hésitent ...](#)

4. Prendre une copie de votre source initial, 2degv1.c et la renommer en 2degcomp.c

La compilation se fera par `gcc -Wall -o 2degcomp 2degcomp.c -lm` (Ne pas oublier -lm c'est un L minuscule, pas un 1 !!)

Faites une recherche des racines dans les complexes. L'idée sera d'afficher les racines sous la forme $c + i * d$ ou $c - i * d$. Le c et d dans ce cas seront le $-b$ (de départ) / $2 * a$ et $d = \sqrt{-\Delta} / 2 * a$. Le terme i sera afficher " en dur " en tant que caractère dans un `printf()`.

Exemple d'usage de votre `printf()` : `printf(" racine x1 =%f + i %f ", c , d) ;`
`printf(" racine x2 =%f - i %f ", c , d) ;`



ANNEXE 1 : Eléments de départ pour la résolution du second degré

```
/******  
/*  
/* NOM      : 2deg.c  
/* TYPE     : Application  
/* SUJET    : calcul resolution second degre  
/*AUTEURS   :  
/* Version  : 1.0  
/*Creation  : /10/2018  
/*Revision  : ..  
/* Acces SRC : ~/src  
/*FABRICATION : gcc -Wall -o <nom_binaire> <nom_source1.c> ..  
/******  
  
#include <stdio.h>  
#include "argv.h"  
int main(int argc, char *argv[])  
{  
    /* déclaration des variables */  
    int a =0 , b =0 , c = 0 ; /* les 3 coefficients entiers */  
    float delta = 0 , x1 = 0, x2 = 0 ; /* le delta et les 2 racines réelles en temps que nombre à virgule */  
    /* message invite utilisateur */  
    printf("*** Entrez 3 coefficients entiers \n) ;  
    /** calcul delta */  
    delta = b *b - 4 * a * c ;  
    if ( delta > 0 )  
    { /* affichage */  
        printf("delta =%f \n",delta);  
        /** TO DO calcul des racines */  
    } else if( )  
    {
```

En-tête standard à renseigner
Et respecter pour chaque TP



```
    /** TO DO **/  
    }  
    else  
}  
    /* fin programme principal **/
```

ANNEXE 2 : Elements de correspondance ALGORITHME et Langage C

Les ALTERNATIVES :

Elle équivaut au SI (condition) ALORS SINON FINSI du pseudo-langage.

Syntaxe : if(<expression_de_test>)

```
{
    bloc d'instructions;          /* NE PAS OUBLIER les accolades d'ouverture / fermeture */
}
else
{
    bloc d'instructions;
}
```

On peut imbriquer les tests pour avoir un si sinon si : if(<test1>){ } else
if(<test2>) {}
else

Les tests à choix multiple

switch(<expression>)

```
{
    case <expression choix1> : [liste 1 d'instructions];
        /* composée de 1 ou plusieurs lignes */
        break;
    case <expression choix2> : [liste 2 d'instructions];
        break;
    .
    default : [liste d'instructions par défaut]; /* non indispensable */
}
```

ANNEXE 3 : Pour l'éditeur VI le minimum vital...

Il faut comprendre qu'il y a plusieurs modes de fonctionnement dont :

Le **mode edition** : vous tapez normalement le texte.

Pour **entrer dans ce mode tapez i** une seule fois.

Pour sortir de ce mode et **passer en mode commande tapez ESC puis :**

w nom_fichier pour enregistrer votre texte sous nom_fichier.

wq pour quitter

e nom_fichier pour (re)entrer dans l'editeur.

Dans le mode commande on efface du texte par x (un seul caractere) , n dd (efface n lignes)

G en fin de fichier , U en debut de page etc.

.

