

Langage C
TP7
 Itheri Yahiaoui && Stéphane Cormier

Exercice 1 : Les structures avec des attributs statiques

- Soit l'exemple suivant :

Test1.c	complex.c	complex.h
<pre>#include "complex.h" int main(void) { complex n1, n2, res; n1 = lecture_complex(); printf("n1= "); affichage_complex(n1); n2 = lecture_complex(); printf("n2= "); affichage_complex(n2); res = addition_C(n1,n2); printf("La somme de n1 et n2 est égale à :"); affichage_complex(res); return 0; }</pre>	<pre>#include "complex.h" complex lecture_complex() { complex tmp; printf("Donner la partie réelle:"); scanf("%f",&tmp.reel); printf("donner la partie imaginaire:"); scanf("%f",&tmp.imag); return (tmp); } void affichage_complex(complex c) { printf("%.2f + %.2fi\n",c.reel, c.imag); } complex addition_C(complex c1, complex c2) { complex tmp; tmp.reel = c1.reel + c2.reel; tmp.imag = c1.imag + c2.imag; return (tmp); }</pre>	<pre>#ifndef COMPLEX_H #define COMPLEX_H #include <stdio.h> typedef struct complex { float reel; float imag; } complex; complex lecture_complex(); void affichage_complex(complex); complex addition_C(complex, complex); #endif /* COMPLEX_H */</pre>

- Compiler et exécuter votre code, vous pouvez tester les trois possibilités de compilation

<pre>\$> gcc complex.c Test1.c \$> ./a.out</pre>	<pre>\$> gcc -o Test1 complex.c Test1.c \$> ./Test1</pre>	<pre>\$> gcc -c complex.c \$> gcc -c Test1.c \$> gcc -o Test1 complex.o Test1.o \$> ./Test1</pre>
--	---	---

- Compléter votre bibliothèque en ajoutant les fonctions dont les prototypes sont
 - void saisir(complex *pc);
 - void ajout(complex *pc1, complex c2);
 - complex produit_C (complex c1, complex c2);
 - void produit(complex *pc1, complex c2);
 - complex Conjugue_C(complex c) ;
 - complex Quotient_C(complex c1, complex c2) ;
 - double Module_C(complex c) ;
 - double Argument_C(complex c) ;
 - complex Rac2C(complex z) ; calcule la racine tel que $r^2=z$;

- Créer un tableau dynamique comportant N nombres de type complex. N et les nombres sont saisis par l'utilisateur.
- Calculer le module et l'argument de chacun des nombres stockés.

Exercice 2 : Les structures avec des attributs statiques et dynamiques

- Soit la structure suivante

```
struct Etudiant {
    char *nom ;
    char *prenom ;
    int  num_etud ;
    int  age_etud ;
    float Notes[10] ;
    float moyenne ;
};
```

- Ecrire un programme qui crée un tableau dynamique d'étudiants, et qui affiche à l'utilisateur le menu suivant :
 - a) Saisir le nombre d'étudiants.
 - b) Saisir les données des étudiants.
 - c) Afficher le contenu du tableau trié selon le nom.
 - d) Afficher les données du Major.
 - e) Afficher la liste des étudiants ayant une moyenne supérieure ou égale à dix.
 - f) Afficher le contenu du tableau trié selon la moyenne.
- Modifier le programme précédent en ajoutant à la structure un champ de type « Etud_adresse », telle que « Etud_adresse » est une autre structure définie comme suit :

```
struct Etud_adresse
{
    int numero ;
    char *rue ;
    char *cdp ;
    char *ville ;
    char *pays ;
};
```