**CHAP IV** : LA PROGRAMMATION MODULAIRE OU SOUS-PROGRAMME

I .Fonction

1.syntaxe (voir cours).

2.exemple :

a .créer une fonction qui prend en paramètre deux entiers et retourne leur somme.

Int somme (int a, int b){

Int som ;

Som = a+b ;

Return som ;

}

b.créer une fonction qui retourne la somme de deux entiers.

Int somme(){

Int a=5, b=10 ;

Return a+b ;

}

Exemple : Procedure .

Void direbonjour(){

Printf(« Bonjour ») ;

}

direBonjour() ;

POUR APPLER UNE FONCTION IL FAUT LE FAIRE SOIT :

* Dans une fonction d’affichage.

EXP : printf(‘’ la somme est %d ‘’,somme()) ;

* Dans une expression d’affectation.

Int som= somme(2,4)

* Dans une expression de calcul

Prod = somme(2,4)\*2 ;

* Dans une condition

If(somme(2,4)>=15)

{

Puts(‘’ok’’) ;

}

Else

{

Puts(‘’ko’’) ;

}

EXERCICE D’APPLICATION

Créer les modules suivants :

-void menu (void)

-int addition (int,int)

-int difference (int,int)

-Int multiplication (int,int)

-int division (int,int)

-int modulo(int,int)

**Solution**

**1ere méthode**