

Les fonctions

- ✓ Utilisation de fonctions justifiée par
 - » Répétition d'un même traitement
 - » Nécessité de décomposer un programme en modules effectuant un travail bien précis pour réduire la complexité
 - » Besoin de définir des fonctions n'existant pas dans la version standard
 - » Plus de clarté
 - Programme découpé en parties
 - Objectifs (rôles) bien définis
- ✓ Fonction main
 - » Fonction principale de tout programme
 - » C'est elle qui appelle directement ou indirectement toutes les autres fonctions
 - » Elle renvoie par convention un entier

Les fonctions

- ✓ Déclaration d'une fonction
 - » Syntaxe

```
<type_resultat> nom_fonction (type1 argument1, type2 argument2...)  
{  
    // corps de la fonction  
    return resultat; //si <type_resultat> différent de void  
}
```

- Une fonction ne peut retourner qu'un seul résultat mais peut admettre plusieurs arguments (paramètres)
- Si la fonction ne retourne rien, le mot clé "void" devra être mis à la place du <type_resultat>
- Par défaut, <type_resultat> = int
- <type_resultat> : scalaire ou pointeur

```
#include <stdio.h>  
#include <math.h>  
  
float surfaceDisque (int r)  
{  
    return (r*r*M_PI);  
}
```

Les fonctions

- ✓ Appel à une fonction
 - » La déclaration d'une fonction doit précéder son utilisation
 - » Si impossibilité de déclarer une fonction avant de l'appeler
 - Déclarer le prototype de la fonction avant de faire appel à la fonction
 - Le prototype (profil, signature) d'une fonction correspond simplement à la première ligne de la déclaration de cette fonction terminée par un point-virgule
 - » Dans la définition d'une fonction
 - Les arguments sont appelés les paramètres formels
 - » A l'appel d'une fonction
 - Les arguments qui lui sont transmis sont appelés les paramètres effectifs (réels)
 - » Les paramètres effectifs et les paramètres formels peuvent ne pas avoir le même nom mais ils doivent correspondre en nombre et en type

Les fonctions

- » Supposons la fonction $f(x) = a * x + b$
 - En mathématiques
 - x est appelée "variable" de la fonction
 - a et b sont appelées "paramètres" de la fonction
 - En informatique
 - x est appelée "paramètre formel" de la fonction
 - a et b sont appelées "variables locales" de la fonction
 - La donnée qui donne sa valeur à x est appelée "paramètre effectif (réel)"