

Support de cours

Cours:

Initiation à la programmation (en C++)

Vidéo:

Introduction (partie 3)

Concepts (extraits des sous-titres générés automatiquement) :

Programmeur-concepteur. Programmeur-utilisateur. Fonction score. Portion de programme réutilisable. Prototype de la fonction. Deuxième aspect. Dernier aspect. Fonction. Utilisation d'une fonction. Ensemble des variables extérieures. Prototype. Création effective de la fonction. Portion de programme. Type de la valeur. Corps de la fonction.



[vers la recherche de séquences vidéo](#)
(dans Initiation à la programmation (en C++).)



[vers la vidéo](#)

Center for Digital Education. Plus de matériel de soutien pédagogique ici :

<https://www.epfl.ch/education/educational-initiatives/cede/educational-technologies-gallery/boocs-en/>
page 1/7

Fonctions : introduction

(Partie 3)

Initiation à la programmation (C++)

Vincent Lepetit, Jean-Cédric Chappelier et Jamila Sam

...

notes

résumé

0m 0s



fonction = portion de programme réutilisable ou importante en soi

caractérisée par :

un corps : la portion de programme à réutiliser ou mettre en évidence ;

un nom : par lequel on désignera cette fonction ;

des paramètres : ensemble de variables extérieures à la fonction dont le corps dépend pour fonctionner ;

un type et une valeur de retour : ce que la fonction renvoie au reste du programme

L'utilisation de la fonction dans une autre partie du programme se nomme un **appel** de la fonction.

Une fonction est donc une portion de programme réutilisable, comme la fonction saisiu>et/u>calcul ou importante en soi comme la fonction score. Elle est caractérisée par un corps, qui est la portion de programme réutilisée ou à mettre en évidence. Elle est aussi caractérisée par son nom qui va permettre de faire référence à cette fonction. Elle peut avoir des paramètres, qui sont l'ensemble des variables extérieures à la fonction dont le corps a besoin pour fonctionner. La fonction peut également renvoyer une valeur en retour et cette valeur doit avoir un type. Et finalement, l'utilisation d'une fonction dans le reste du programme

notes

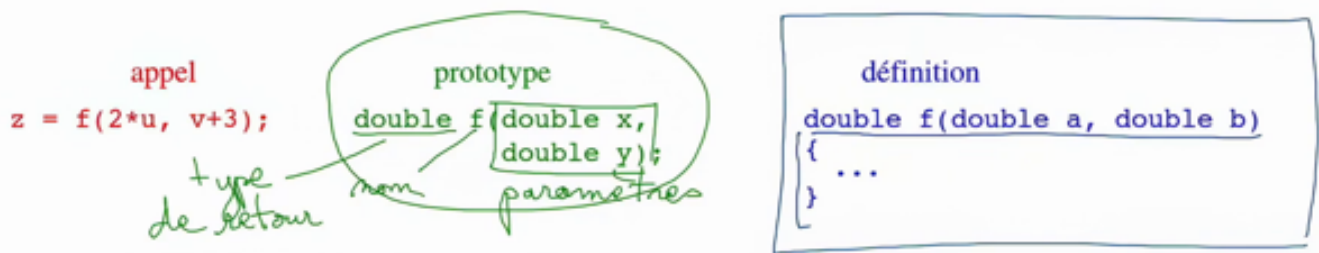
résumé

0m 1s



Les « 3 facettes » d'une fonction

- ▶ Résumé / Contrat (« prototype »)
- ▶ Création / Construction (« définition »)
- ▶ Utilisation (« appel »)



se nomme un appel à la fonction. Une fonction a donc trois facettes. Tout d'abord son prototype, qu'on peut voir comme un résumé de ce que doit faire la fonction. Puisqu'il contient son nom, ses paramètres, qui correspondent aux valeurs dont a besoin la fonction pour pouvoir fonctionner. Ainsi que le type de la valeur que va fournir la fonction. Un deuxième aspect est la création effective de la fonction ou sa définition. Qui commence comme un prototype mais qui contient également

notes

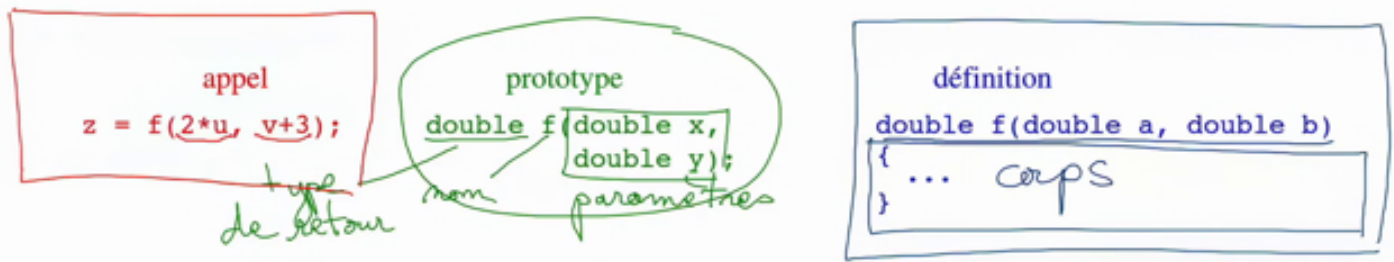
résumé

0m 49s



Les « 3 facettes » d'une fonction

- ▶ Résumé / Contrat (« prototype »)
- ▶ Création / Construction (« définition »)
- ▶ Utilisation (« appel »)



ce qu'on appelle le corps de la fonction, qui est le code qui va être exécuté quand on utilise la fonction. Le dernier aspect est donc l'utilisation de la fonction ou son appel. On va utiliser la fonction en lui donnant des valeurs effectives pour ces paramètres.

notes

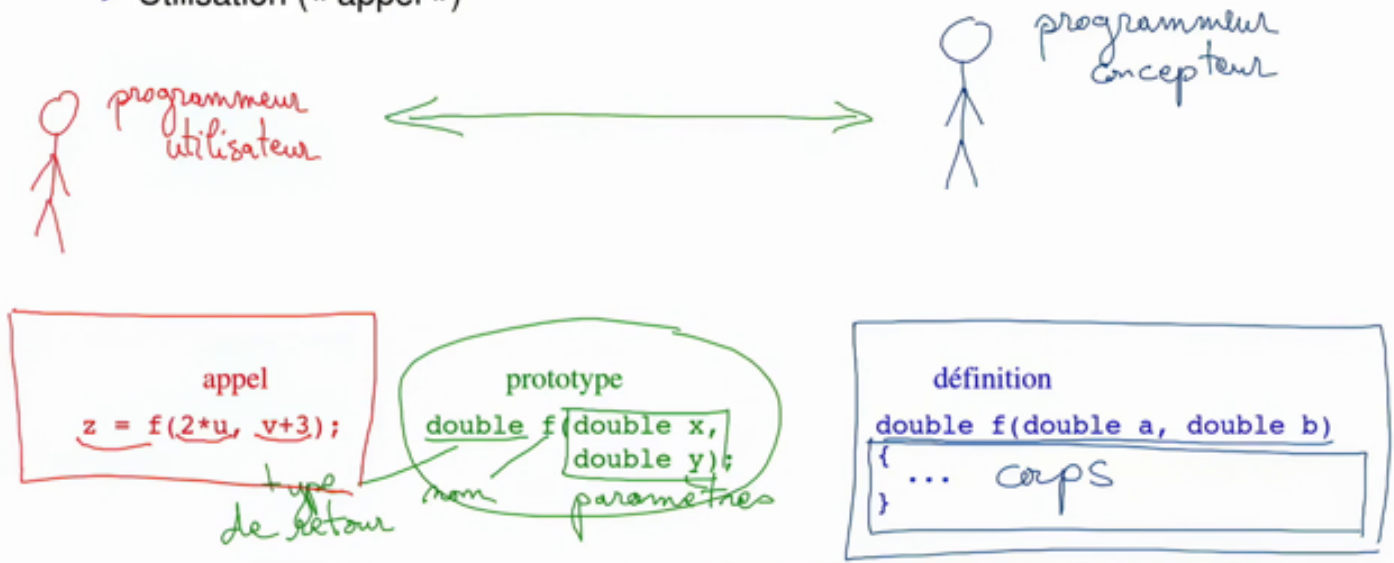
résumé

1m 37s



Les « 3 facettes » d'une fonction

- Résumé / Contrat (« prototype »)
- Création / Construction (« définition »)
- Utilisation (« appel »)



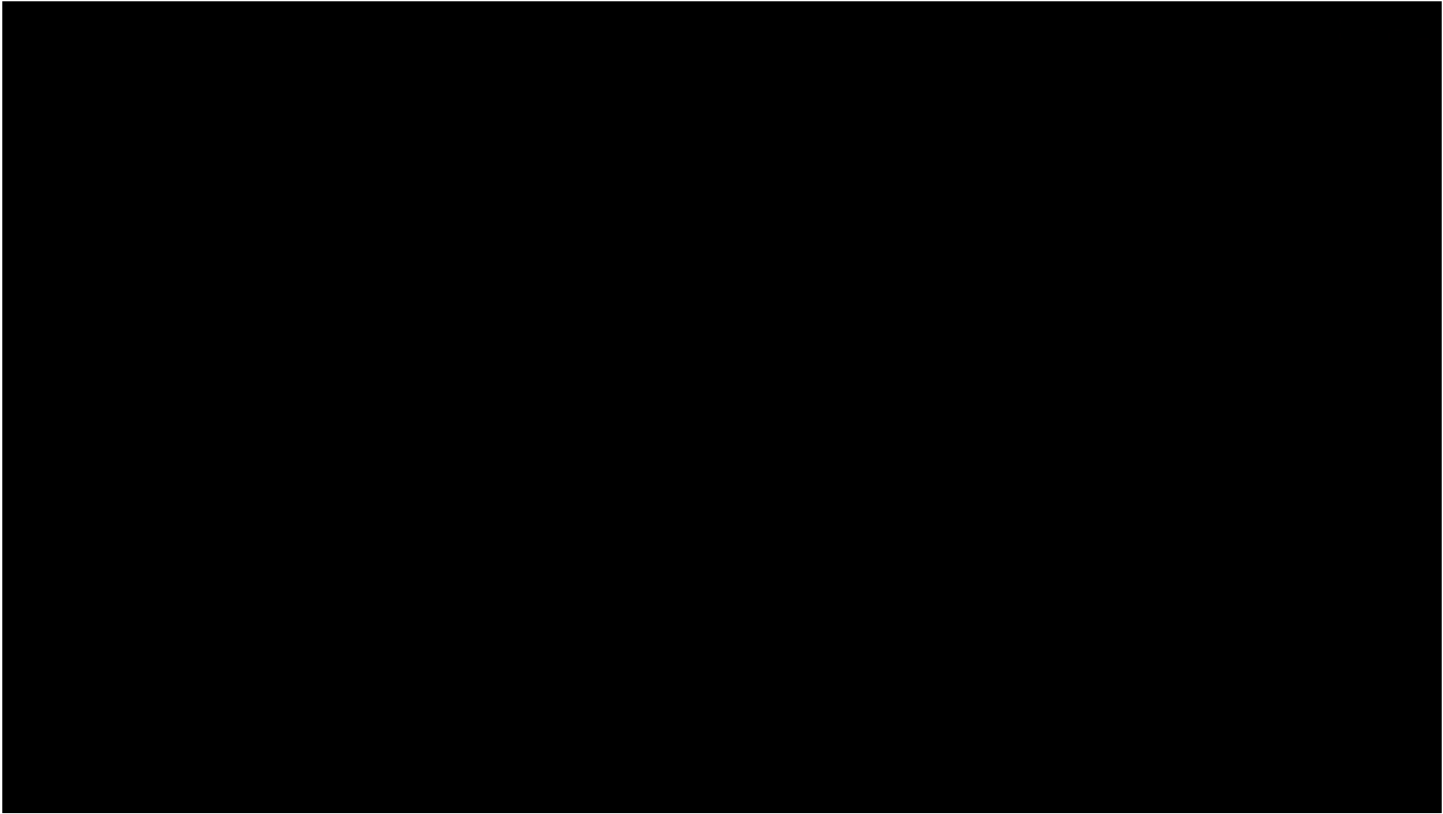
La fonction va fournir une valeur qu'on va pouvoir affecter à une variable. En pratique le programmeur-concepteur, c'est à dire la personne qui va écrire la définition de la fonction, n'est pas forcément la même personne que le programmeur-utilisateur, c'est à dire la personne qui va utiliser la fonction. Le programmeur-utilisateur n'a pas à connaître le corps de la fonction. Tout ce dont il a besoin de connaître c'est le prototype de la fonction, pour pouvoir l'appeler.

notes

résumé

2m 1s





Le prototype sert donc d'accord entre le programmeur-utilisateur et le programmeur-concepteur. Le programmeur-utilisateur doit respecter le prototype de la fonction quand il appelle la fonction. Le programmeur-concepteur va s'arranger pour que le prototype corresponde au problème que le programmeur-utilisateur veut résoudre en appelant la fonction. en appelant la fonction.

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

.....

.....

.....

.....

.....

2m 49s

