

Programmeren in C

Praktijkcursus van 5 days - 35h

Ref : LGC - Prijs 2026 : € 2 610 excl. BTW

Deze intensieve cursus stelt u in staat om een echt operationele kennis van de taal te verwerven. Er wordt uitgelegd hoe de verschillende mechanismen werken en aan de hand van een aantal praktische oefeningen wordt getoond hoe u ze kunt implementeren. Aan het eind van de cursus ben u in staat om robuuste en portable C-programma's te schrijven.

PEDAGOGISCHE DOELSTELLINGEN

Na afloop van de opleiding kan de cursist:

De productieketen van een programma geschreven in C onder de knie krijgen

De operatoren, expressies en controlestructuren van de C-taal gebruiken

Datastructuren, arrays, pointers en tekenreeksen manipuleren

Programmacode organiseren met functies

Gebruik de belangrijkste standaard C-bibliotheken

HANDS-ON WORK

Deelnemers krijgen de beschikking over Unix of Windows (PC) machines zodat ze de gepresenteerde concepten in de praktijk kunnen brengen.

HET PROGRAMMA

laatste update: 02/2025

1) Eerste stappen in C

- Inleiding tot de taal C en de voordelen ervan.
- C++ versus C. C++11 en C11 standaarden.
- Bronbestanden (.c, .h).
- Algemene structuur van een programma.
- De basissyntaxis van de taal.
- Gegevenstypes en basisconstanten.
- Globale en lokale variabelen.
- Parameters opslaan en doorgeven.
- Opgemaakte invoer/uitvoer.
- De opmerkingen.
- Basisgebruik van de productielijn.
- Bewerkings-, compilatie- en uitvoeringsomgevingen.
- Voer een initieel programma uit.

2) Operatoren en uitdrukkingen

- Rekenkundige operatoren.
- Mechanismen voor het evalueren van uitdrukkingen.
- Post en pre-increment of decrement.
- Precedentie en associativiteit van operatoren.
- Toewijzingsoperatoren.
- Hoe logische expressies werken.
- Logische uitdrukkingen in while, if... statements
- Vergelijkingsoperatoren : <, >, ==, !=...

DEELNEMERS

Ontwikkelaars, ingenieurs, projectmanagers die dicht bij de ontwikkeling staan.

VOORAFGAANDE VEREISTEN

Basisvaardigheden in programmeren.

VAARDIGHEDEN VAN DE CURSUSLEIDER

De deskundigen die de cursus leiden zijn specialisten op het betreffende vakgebied. Zij werden geselecteerd door onze pedagogische teams zowel om hun vakkennis als hun pedagogische vaardigheden voor elke cursus die zij geven. Zij hebben minstens vijf tot tien jaar ervaring in hun vakgebied en oefenen of oefenden verantwoordelijke bedrijfsfuncties uit.

BEOORDELINGSMODALITEITEN

De cursusleider beoordeelt de pedagogische vooruitgang van de deelnemer gedurende de gehele cursus aan de hand van meerkeuzevragen, praktijksituaties, praktische opdrachten, ... De deelnemer legt ook van tevoren en naderhand een test af ter bevestiging van de verworven kennis.

PEDAGOGISCHE EN TECHNISCHE MIDDELEN

- De gebruikte pedagogische middelen en cursusmethoden zijn voornamelijk: audiovisuele hulpmiddelen, documentatie en cursusmateriaal, praktische oefeningen en correcties van de oefeningen voor praktijkstages, casestudies of reële voorbeelden voor de seminars.
- Na afloop van de stages of seminars verstrekt ORSYS de deelnemers een evaluatievragenlijst over de cursus die vervolgens door onze pedagogische teams wordt geanalyseerd.
- Na afloop van de cursus wordt een presentielijst per halve dag verstrekt, evenals een verklaring van de afronding van de cursus indien de stagiair alle sessies heeft bijgewoond.

TOEGANGSMODALITEITEN EN -TERMIJNEN

De inschrijving dient 24 uur voor aanvang van de cursus plaatsgevonden te hebben.

TOEGANKELIJKHEID VOOR MINDERVALIDEN

Is voor u speciale toegankelijkheid vereist? Neem contact op met mevr. FOSSE, contactpersoon voor mindervaliden, via het adres psh-accueil@ORSYS.fr om uw verzoek en de haalbaarheid daarvan zo goed mogelijk te bestuderen.

- Logische operatoren: AND, OR, negatie.
- Samengestelde numerieke types. Conversieregel in gemengde uitdrukkingen. Impliciete/expliciete conversies.
- Variabele initialisatie.
- Rekenen met adressen.
- Invoer-/uitvoerformaten gekoppeld aan digitale types.
- Bitwise operatoren: AND, OR, exclusieve OR, complement tot 1, negatie. Shift-operatoren: >>, <<.
- Voorwaardelijke uitdrukking met de ternaire operator.

Operatoren en uitdrukkingen gebruiken.

3) Besturingsstructuren

- Begrip van blokken.
- Lusstructuren: while, for.
- Luscontrole-instructies: break, continue.
- Keuzestructuren: als, anders, anders als.
- Meerkeuzestructuur: omschakelen.

Implementatie van controlestructuren.

4) Arrays, pointers en tekenreeksen

- De elementen van een array definiëren, initialiseren en openen.
- Een pointer definiëren. Het geheugenadres van een object ophalen. De inhoud van een pointer benaderen.
- Pointer/tabel equivalenten.
- Wijzers berekenen.
- Tekenreeksen.
- Voorbeelden van stringmanipulatie.
- C11 Unicode-tekenreeksen.

Het manipuleren van arrays, pointers en tekenreeksen.

5) De structuren

- Belangen van de structuren.
- Velden in een structuur declareren, initialiseren en openen.
- Gebruik geneste structuren.
- Nieuwe types maken met Typedef.
- Bitvelden.
- Vakbonden.
- Opsommingen.
- Anonieme structuren en opsommingen in C11.
- Wijzers naar structuren definiëren.

Implementatie van nieuwe gegevensstructuren.

6) De functies

- Definitie van een functie.
- Roep een functie aan.
- Parameters doorgeven: met waarde of door verwijzing.
- De retourcode van een functie. Typen terugkeer.
- De "hoofd"-functie.

Uw code opsplitsen met behulp van functies. Functie-aanroepen beheren.

7) Aparte compilatie, toewijzingsklasse

- Bedieningsmechanisme productielijn.
- Gebruik van bronbibliotheken.
- Notie van Makefile.
- Geheugenconfiguratie van een C-programma (stack, heap, etc.).
- Klassen voor toewijzing van variabelen (auto, register, static, extern).
- Verschillende scenario's voor afzonderlijke compilatie.

- Notie van extern object.
- Globale en statische gegevens.
- Lokale gegevens.
- Zichtbaarheidsregel.
- Aanvullende informatie over functies en initialisaties.

8) De preprocessor

- Gebruik voorgedefinieerde macro's (symbolische constanten). u eigen macro's definiëren met `#define`.
- Macro's definiëren als functies. `#` en `##` markeringen gebruiken.
- Annuleer de constante definitie met `#undef`.
- Voorwaardelijke compilatie: `#if`, `#ifdef`, `#ifndef`, `#elif`, `#endif`.
- Neem bronnen op met `#include`.

Preprocessor directives gebruiken. Voorwaardelijke compilatie instellen.

9) Standaardbibliotheken

- Wiskundige rekenfuncties (`sqrt`, `sin`...).
- Invoer-/uitvoerfuncties (`fprintf`, `fscanf`...).
- Functies voor bestandstoegang (`fread`, `fwrite`...).
- Functies voor het manipuleren van strings (`strlen`, `strcat`...).
- Geheugenbeheerfuncties (`malloc`, `free`...).
- Het opzetten van geketende structuren (geketende lijsten, n-ary bomen, enz.).
- De "veilige" functies in de standaard C11 bibliotheek (`strcat_s`, `strlen_s`, etc.).

Gebruik van de belangrijkste standaardbibliotheekfuncties.

DATA

Neem contact met ons op