

Opleiding : WebAssembly, verhoogt de prestaties van uw webapplicaties

Binary in de motor van uw webbrowsers stoppen

Praktijkcursus - 3d - 21u00 - Ref. WAY

Prijs : 1830 € V.B.

★★★★★ 5 / 5

WebAssembly (WASM), een officiële W3C-standaard sinds 2019, maakt het mogelijk om supersnelle en ultralichte toepassingen op het web te schrijven. Deze toepassingen kunnen al worden geschreven in allerlei bestaande brontalen: C/C++, Rust, Go, Java, enz. Ze moeten alleen nog worden geporteerd voor toegang in een browser of beveiligde container. Deze hands-on training biedt de sleutels tot het ontwikkelen van WASM-code en het compileren van bestaande programma's in WebAssembly.

Pedagogische doelstellingen

Aan het einde van de training is de deelnemer in staat om:

- ✓ Inzicht in de architectuur en omgeving van de W3C WebAssembly-standaard
- ✓ Beheers de binaire instructieset van de WASM-taal en de tekstuele weergave ervan, de WAT-indeling.
- ✓ De JavaScript API gebruiken voor interactie met WASM-modules
- ✓ Weten hoe je C/C++ compileert met de Emscripten suite
- ✓ Ontwikkelen met de taal AssemblyScript
- ✓ Een C/C++-programma of -bibliotheek overzetten naar WASM

Doelgroep

Ontwikkelaars, webontwikkelaars, integrators, technische architecten, managers van technische oplossingen.

Voorafgaande vereisten

Basiskennis van HTML, talen zoals JavaScript en C, en commandotalen zoals shell, Bash of CMD (DOS).

Opleidingsprogramma

DEELNEMERS

Ontwikkelaars, webontwikkelaars, integrators, technische architecten, managers van technische oplossingen.

VOORAFGAANDE VEREISTEN

Basiskennis van HTML, talen zoals JavaScript en C, en commandotalen zoals shell, Bash of CMD (DOS).

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.

1 Inleiding tot WASM

- Welke problemen pakt WebAssembly aan?
- Geschiedenis van WASM.
- Architectuur.
- Draagbaarheid, veiligheid, prestaties.
- Organisatie van de specificatie.
- Documentatie.
- WASI, Bytecode Alliance.
- Modulestructuur.

Praktisch werk

Eenvoudige modules schrijven in WASM met WAT. Compileren en uitvoeren met wat2wasm en node.

2 WAT tekstuele taal

- Beschrijving van een WAT-ontwikkelomgeving.
- Visual Studio Code-uitbreidingen.
- De verschillende moduleverklaringen.
- Opmerkingen, S-uitdrukkingen.
- Functies en de instructiestapel.
- Een functie of ander artefact importeren/exporteren.
- Globale objecten.
- Lineair geheugen.
- Wijzertabellen.
- De verschillende instructies: lus, voorwaarden, bewerkingen, val.
- Startfunctie "start".
- De JavaScript-interface voor het gebruik van een WASM-module.

Praktisch werk

Schrijf en compileer een module in WAT die een paar wiskundige basisfuncties biedt (factorial, Fibonacci, enz.). Voer het bestand uit in Node en in een browser.

3 Runtimes WASM

- Runtime-vereisten.
- Lijst met runtimes.
- Beschrijving van WASI.
- Runtime-installatie.
- WASM-programma's uitvoeren met runtimes.

Praktisch werk

Een eenvoudig programma geschreven in Rust compileren in WASM en uitvoeren op verschillende runtimes (wasm3, wasmtime, etc.)

4 AssemblyScript

- Installatie van de Node AssemblyScript module.
- Een project starten met asinit.
- Vuilnisman en geheugen.
- Programmeren met objecten.
- Integratie van een WASM-bibliotheek gemaakt in AssemblyScript.

Praktisch werk

Het schrijven van een WASM-module in AssemblyScript die de kleuren van de punten van een Mandelbrot-fractal berekent en deze bibliotheek integreert in een frontend die de fractal weergeeft.

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.

5 Het hulpprogramma Emscripten

- Talen die kunnen worden overgezet naar WASM.
- Algemene presentatie van Emscripten.
- Historische achtergrond.
- Officiële installatie.
- Installatie onder Debian/Ubuntu met apt.
- De emcc compiler.
- Het JavaScript-envelopbestand.
- Compilatieopties.
- Belstrategieën vanuit JavaScript (ccall, cwrap, etc.).

Praktisch werk

Schrijf een eenvoudig programma in C, compileer het in WASM en gebruik het met Node en in een browser.

6 Boekenwinkel portier

- Compileren en configureren met Emscripten en Autoconf.
- Compileren en configureren met Emmake en Emconfigure.
- Interactie met makefiles.
- MODULARIZE, EXPORTED_FUNCTIONS, EXPORTED_RUNTIME_METHODS compilatieopties.
- Het virtuele bestandssysteem.
- Omgevingsvariabelen.

Data en plaats

KLAS OP AFSTAND

2026 : 13 apr., 24 juni, 16 nov.

PARIS LA DÉFENSE

2026 : 30 maa., 17 juni, 2 nov.