



Szkolenie PI-C-02 C Essentials - Part 2 (Intermediate)

Numer usługi 2024/05/22/142469/2158470

6 273,00 PLN brutto

5 100,00 PLN netto

179,23 PLN brutto/h

145,71 PLN netto/h

SOFTRONIC

SPÓŁKA Z

OGRANICZONĄ

ODPOWIEDZIALNOŚĆ

CIA



📍 zdalna w czasie rzeczywistym

👤 Usługa szkoleniowa

🕒 35 h

📅 26.08.2024 do 30.08.2024

Informacje podstawowe

Kategoria	Informatyka i telekomunikacja / Programowanie
Sposób dofinansowania	wsparcie dla osób indywidualnych wsparcie dla pracodawców i ich pracowników
Grupa docelowa usługi	Szkolenie PI-C-02 C Essentials - Part 2 (Intermediate) jest dedykowane programistom, którzy chcą poszerzyć swoje umiejętności w języku programowania C, skupiając się na zaawansowanych aspektach składni, semantyki, zarządzania pamięcią, strukturach, plikach i strumieniach. Zalecane jest dla osób aspirujących do kariery w programowaniu oprogramowania, programowaniu sieciowym oraz tworzeniu systemów operacyjnych, które posiadają już podstawową wiedzę na temat składni języka C lub ukończyły szkolenie C Essentials - Part 1.
Minimalna liczba uczestników	3
Maksymalna liczba uczestników	7
Data zakończenia rekrutacji	12-08-2024
Forma prowadzenia usługi	zdalna w czasie rzeczywistym
Liczba godzin usługi	35
Podstawa uzyskania wpisu do BUR	Certyfikat systemu zarządzania jakością wg. ISO 9001:2015 (PN-EN ISO 9001:2015) - w zakresie usług szkoleniowych

Cel

Cel edukacyjny

Celem tego szkolenia jest umożliwienie uczestnikom zdobycia zaawansowanych umiejętności w języku programowania C, skoncentrowanych na bardziej zaawansowanych aspektach składni, semantyki, zarządzania pamięcią, strukturach, plikach, strumieniach oraz najczęściej używanych funkcjach bibliotecznych. Szkolenie ma na celu przygotowanie aspirujących programistów do zaawansowanego programowania oprogramowania, programowania sieciowego i tworzenia systemów operacyjnych.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Rozumie podstawowe pojęcia dotyczące języków programowania	wyjaśnia różnicę między językami naturalnymi a sztucznymi, opisuje języki maszynowe oraz wysokiego poziomu, wyjaśnia proces kompilacji oraz uzyskiwania kodu maszynowego.	Test teoretyczny
Potrafi pisać proste programy w języku C	definiuje i używa zmienne, stosuje literały całkowitoliczbowe oraz zmiennoprzecinkowe, używa operatorów arytmetycznych oraz instrukcji warunkowych (if, else), korzysta z funkcji printf() i scanf() do wejścia i wyjścia danych.	Test teoretyczny
Zna i stosuje różne typy danych oraz operatory w języku C	rozdziela typy int, float, char oraz ich literały, stosuje operatory arytmetyczne, porównania, logiczne oraz bitowe, rozumie priorytet i wiązanie operatorów, stosuje post- i preinkrementację oraz dekrementację.	Test teoretyczny
Potrafi kontrolować przepływ programu	używa pętli (while, do, for) do iteracji, stosuje instrukcje break i continue do kontroli pętli, implementuje konwersje i rzutowanie typów, rozumie działanie instrukcji switch.	Test teoretyczny
Umie definiować i manipulować tablicami	deklaruje i inicjalizuje tablice jedno- i wielowymiarowe, stosuje wskaźniki do operacji na tablicach, implementuje sortowanie tablic, używa wskaźników do pracy ze stringami.	Test teoretyczny

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Zarządza pamięcią dynamicznie w języku C	stosuje funkcje malloc() i free() do alokacji i zwalniania pamięci, rozumie zagrożenia związane z niewłaściwym zarządzaniem pamięcią, używa wskaźników prostych i wskaźników NULL, stosuje operator sizeof.	Test teoretyczny
Potrafi definiować i używać struktur danych	deklaruje, inicjalizuje i używa struktur, stosuje wskaźniki do struktur oraz tablice struktur, rozumie podstawy rekurencyjnych kolekcji danych.	Test teoretyczny
Definiuje i wywołuje funkcje w języku C	deklaruje, definiuje i wywołuje funkcje, stosuje zmienne lokalne i parametry funkcji, przekazuje wskaźniki, tablice i struktury jako parametry funkcji, używa funkcji z typem void oraz deklaratorem extern.	Test teoretyczny
Operuje na plikach i strumieniach danych	rozdziela pliki i strumienie, otwiera i zamyka strumienie, używa zmiennej errno, czyta i zapisuje dane przy użyciu funkcji fgetc(), fputc(), fgets() i fputs(), stosuje surowe wejście/wyjście za pomocą fread() i fwrite().	Test teoretyczny
Rozumie działanie preprocesora i potrafi tworzyć złożone deklaracje	używa dyrektyw preprocesora #include, #define, #undef, stosuje makra proste i parametryzowane, implementuje kompilację warunkową przy użyciu dyrektyw #if i #ifdef, definiuje typy zdefiniowane przez użytkownika oraz wskaźniki do funkcji.	Test teoretyczny

Kwalifikacje

Inne kwalifikacje

Uznane kwalifikacje

Pytanie 5. Czy dokument jest certyfikatem, dla którego wypracowano system walidacji i certyfikowania efektów uczenia się na poziomie międzynarodowym?

Tak

Informacje

Podstawa prawna dla Podmiotów / kategorii Podmiotów	uprawnionych do wydawania dokumentów potwierdzających uzyskanie kwalifikacji, w tym w zawodzie
Nazwa/Kategoria Podmiotu prowadzącego walidację	Pearson VUE
Podmiot prowadzący walidację jest zarejestrowany w BUR	Nie
Nazwa/Kategoria Podmiotu certyfikującego	Open EDG
Podmiot certyfikujący jest zarejestrowany w BUR	Nie

Program

Szkolenie **PI-C-02 C Essentials - Part 2 (Intermediate)** jest zaprojektowane dla uczestników, którzy aspirują do zaawansowanej wiedzy w języku programowania C, skupiając się na bardziej zaawansowanych aspektach składni, semantyki i praktyk programistycznych. Uczestnicy zdobędą umiejętności związane z zarządzaniem pamięcią, strukturami danych, plikami i strumieniami, a także opanują najczęściej używane funkcje biblioteczne w języku C. Szkolenie jest zalecane dla osób zainteresowanych karierą w programowaniu oprogramowania, programowaniu sieciowym i tworzeniu systemów operacyjnych, wymagając od uczestników podstawowego zrozumienia składni języka C, a korzystanie z kursu C Essentials - Part 1 jest korzystne.

Uczestnicy nauczą się zaawansowanych zagadnień, takich jak manipulacja wskaźnikami, obsługa plików, zarządzanie pamięcią, a także korzystanie z preprocesora i deklaracji. Kurs obejmuje także tematy związane z funkcjami, strukturami, operacjami na stringach oraz analizą i tworzeniem złożonych deklaracji. Po ukończeniu szkolenia, uczestnicy będą gotowi do bardziej zaawansowanych projektów programistycznych w języku C i dalszego rozwoju w obszarze programowania oprogramowania oraz tworzenia systemów operacyjnych.

Szkolenie składa się z wykładu wzbogaconego o prezentację. W trakcie szkolenia każdy Uczestnik wykonuje indywidualne ćwiczenia - laboratoria, dzięki czemu zyskuje praktyczne umiejętności. W trakcie szkolenia omawiane jest również studium przypadków, w którym Uczestnicy wspólnie wymieniają się doświadczeniami. Nad case-study czuwa autoryzowany Trener, który przekazuje informację na temat przydatnych narzędzi oraz najlepszych praktyk do rozwiązania omawianego zagadnienia.

Aby Uczestnik osiągnął zamierzony cel szkolenia niezbędne jest wykonanie przez niego zadanych laboratoriów. Pomocne będzie również ugruntowanie wiedzy i wykonywanie ćwiczeń po zakończonej usłudze. Każdy Uczestnik dysponuje dostępem do laboratoriów przez okres 180 dni.

Który egzamin potwierdza zdobyte umiejętności?

CLA – C Programming Language Certified Associate

Szkolenie trwa 40 godzin zegarowych i jest realizowane w ciągu 5 dni.

W trakcie każdego dnia szkolenia przewidziane są dwie krótkie przerwy "kawowe" oraz przerwa lunchowa.

Program szkolenia

Wprowadzenie

języki: naturalne i sztuczne

języki maszynowe

języki programowania wysokiego poziomu

uzyskiwanie kodu maszynowego: proces kompilacji

pisanie prostych programów

zmienne

wartości całkowite w życiu codziennym i w C

literały całkowitoliczbowe

Typy danych

wartości zmiennoprzecinkowe w życiu codziennym i w C

literały float

operatory arytmetyczne

priorytet i wiązanie

post- i preinkrementacja i dekrementacja

operatory typu op=

typ char i kod ASCII

literały typu char

równoważność danych int i char

operatory porównania

wykonywanie warunkowe i instrukcja if

funkcje printf() i scanf()

Kontrola przepływu

wykonywanie warunkowe: gałąź "else"

typy int i float

konwersje

rzutowanie typów i jego operatory

pętle - while, do i for

kontrola wykonania pętli - instrukcje break i continue

operatory logiczne i bitowe

Tablice

instrukcja switch: różne oblicza "if"

tablice (wektory)

sortowanie w życiu codziennym i w pamięci komputera

inicjatory

wskaźniki

adres, referencja, dereferencja i operator sizeof

wskaźnik prosty i wskaźnik na nic (NULL)

operator &

arytmetyka wskaźników

wskaźniki kontra tablice: różne formy tego samego zjawiska

korzystanie ze stringów

podstawowe funkcje do manipulacji łańcuchami znaków

Zarządzanie pamięcią i struktury

instrukcja switch: różne oblicza "if"

tablice (wektory)

sortowanie w życiu codziennym i w pamięci komputera

inicjatory

wskaźniki

adres, referencja, dereferencja i operator sizeof

wskaźnik prosty i wskaźnik na nic (NULL)

operator &

arytmetyka wskaźników

wskaźniki kontra tablice: różne formy tego samego zjawiska

korzystanie ze stringów

podstawowe funkcje do manipulacji łańcuchami znaków

Zarządzanie pamięcią i struktury

indeksowanie tablic

wykorzystanie wskaźników: zagrożenia i wady

typ void

tablice tablic i tablice wielowymiarowe

alokacja i zwalnianie pamięci: funkcje malloc() i free()

tablice wskaźników kontra tablice wielowymiarowe

struktury

deklarowanie, używanie i inicjalizowanie struktur

wskaźniki do struktur i tablic struktur

podstawy rekurencyjnych kolekcji danych

Funkcje

funkcje

jak deklarować, definiować i wywoływać funkcję

zakres zmiennych, zmienne lokalne i parametry funkcji

wskaźniki, tablice i struktury jako parametry funkcji

wynik funkcji i instrukcja return

typ void jako parametr, wskaźnik i wynik

parametryzacja funkcji main

funkcja zewnętrzna i deklarator extern

pliki nagłówkowe i ich rola

Pliki i strumienie

pliki kontra strumienie,

pliki nagłówkowe potrzebne do operacji na strumieniach,

struktura FILE,

otwieranie i zamykanie strumienia, tryby otwarcia, zmienna errno,

czytanie i zapisywanie do/z strumienia,

predefiniowane strumienie: stdin, stdout i stderr,

manipulacja strumieniem: funkcje fgetc(), fputc(), fgets() i fputs(),

surowe wejście/wyjście: funkcje fread() i fwrite().

Preprocesor i złożone deklaracje

preprocesor

#include: jak korzystać z pliku nagłówkowego

#define: proste i parametryzowane makra

dyrektywa #undef

predefiniowane symbole preprocesora

makrooperatory: # i ##

kompilacja warunkowa: dyrektywy #if i #ifdef

unikanie wielokrotnego kompilowania tego samego pliku nagłówkowego

zakresy deklaracji, klasy pamięci

typy zdefiniowane przez użytkownika

wskaźniki do funkcji

analiza i tworzenie złożonych deklaracji

SOFTRONIC Sp. z o. o. zastrzega sobie prawo do zmiany terminu szkolenia lub jego odwołania w przypadku niezbrania się minimalnej liczby Uczestników tj. 3 osób.

Harmonogram

Liczba przedmiotów/zajęć: 0

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
Brak wyników.					

Cennik

Cennik

Rodzaj ceny	Cena
-------------	------

Koszt przypadający na 1 uczestnika brutto	6 273,00 PLN
Koszt przypadający na 1 uczestnika netto	5 100,00 PLN
Koszt osobogodziny brutto	179,23 PLN
Koszt osobogodziny netto	145,71 PLN
W tym koszt walidacji brutto	1 476,00 PLN
W tym koszt walidacji netto	1 200,00 PLN
W tym koszt certyfikowania brutto	0,00 PLN
W tym koszt certyfikowania netto	0,00 PLN

Prowadzący

Liczba prowadzących: 0

Brak wyników.

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Każdemu Uczestnikowi zostaną przekazane autoryzowane materiały szkoleniowe.

Warunki techniczne

Szkolenie realizowane jest w formule distance learning - szkolenie **on-line w czasie rzeczywistym**, w którym możesz wziąć udział z każdego miejsca na świecie.

Szkolenie odbywa się za pośrednictwem platformy **Microsoft Teams**, która umożliwia transmisję dwukierunkową, dzięki czemu Uczestnik może zadawać pytania i aktywnie uczestniczyć w dyskusji. Uczestnik, który potwierdzi swój udział w szkoleniu, przed rozpoczęciem szkolenia, drogą mailową, otrzyma link do spotkania wraz z hasłami dostępu.

Wymagania sprzętowe:

- komputer z dostępem do internetu o minimalnej przepustowości 20Mb/s.
- wbudowane lub peryferyjne urządzenia do obsługi audio - słuchawki/głośniki oraz mikrofon.
- zainstalowana przeglądarka internetowa - Microsoft Edge/ Internet Explorer 10+ / **Google Chrome** 39+ (sugerowana) / Safari 7+
- aplikacja MS Teams może zostać zainstalowana na komputerze lub można z niej korzystać za pośrednictwem przeglądarki internetowej

Kontakt



Ewa Kasprzak

E-mail ewa.kasprzak@softronic.pl

Telefon (+48) 618 658 840