



Damian Bartuś  
ProEducato



## Programowanie w języku JAVA. Usługa certyfikowana egzaminem ECCC z programowania na poziomie zaawansowanym. Usługa zawiera egzamin ECCC.

Numer usługi 2024/07/22/167688/2231647

📍 Jasło / stacjonarna

🏠 Usługa szkoleniowa

🕒 50 h

📅 16.09.2024 do 04.11.2024

6 980,00 PLN brutto

6 980,00 PLN netto

139,60 PLN brutto/h

139,60 PLN netto/h

## Informacje podstawowe

<b>Kategoria</b>	Informatyka i telekomunikacja / Programowanie
<b>Sposób dofinansowania</b>	wsparcie dla osób indywidualnych
<b>Grupa docelowa usługi</b>	Grupą docelową są osoby chcące uzyskać wiedzę w dziedzinie programowania. Usługa również adresowana jest dla uczestników projektu „Nowe kwalifikacje i kompetencje drogą do kariery”.
<b>Minimalna liczba uczestników</b>	1
<b>Maksymalna liczba uczestników</b>	20
<b>Data zakończenia rekrutacji</b>	15-09-2024
<b>Forma prowadzenia usługi</b>	stacjonarna
<b>Liczba godzin usługi</b>	50
<b>Podstawa uzyskania wpisu do BUR</b>	Certyfikat VCC Akademia Edukacyjna

## Cel

### Cel edukacyjny

Celem szkolenia jest zdobycie wiedzy i umiejętności z zakresu programowania. Uczestnik po ukończeniu szkolenia przygotowany będzie do samodzielnego tworzenia algorytmów w języku programowania JAVA. Dzięki zdobytej wiedzy

oraz umiejętnościom uczestnik będzie przygotowany do tworzenia i zapisywania algorytmów, interpretowania kodu programu oraz optymalizacji rozwiązań.

## Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Tworzy i zapisuje algorytmy	<ul style="list-style-type: none"> <li>• Formułuje algorytmy rozwiązywania konkretnych problemów programistycznych.</li> <li>• Przechodzi od abstrakcyjnych pomysłów do konkretnego opisu kroków w algorytmie.</li> <li>• Zapisuje algorytmy w formie pseudokodu lub języku programowania.</li> </ul>	Test teoretyczny
Interpretuje kod programu	<ul style="list-style-type: none"> <li>• Potrafi analizować kod źródłowy programu i zrozumieć, co robią poszczególne części programu.</li> <li>• Rozpoznaje zmienne, struktury sterujące i wywołania funkcji w kodzie.</li> <li>• Potrafi wskazać, jakie zadania realizuje kod.</li> </ul>	Test teoretyczny
Optymalizuje rozwiązania	<ul style="list-style-type: none"> <li>• Potrafi dokonywać optymalizacji kodu, aby zwiększyć wydajność programu.</li> <li>• Identyfikuje i eliminuje zbędne operacje lub powtarzający się kod.</li> <li>• Poprawia efektywność algorytmów i struktur danych w programie.</li> </ul>	Test teoretyczny
Tworzy klasy i implementuje obiekty	<ul style="list-style-type: none"> <li>• Tworzy klasy, definiując pola, metody i konstruktory.</li> <li>• Implementuje obiekty za pomocą operatora „New” lub odpowiednich mechanizmów w języku programowania.</li> <li>• Rozumie różnicę między klasą a obiektem oraz potrafi wykorzystywać obiekty w swoim kodzie.</li> </ul>	Test teoretyczny
Pracuje z dziedziczeniem	<ul style="list-style-type: none"> <li>• Zrozumie, jak działa dziedziczenie między klasami.</li> <li>• Przeprowadza przeciążenie metod, w tym konstruktorów, w kontekście dziedziczenia.</li> <li>• Potrafi tworzyć hierarchię klas, wykorzystując dziedziczenie.</li> </ul>	Test teoretyczny
Hermetyzuje klasy i metody	<ul style="list-style-type: none"> <li>• Rozumie znaczenie hermetyzacji w programowaniu.</li> <li>• Wykorzystuje odpowiednie modyfikatory hermetyzacji, takie jak public, private, czy protected.</li> <li>• Potrafi kontrolować dostęp do pól i metod klasach.</li> </ul>	Test teoretyczny

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Implementuje polimorfizm	<ul style="list-style-type: none"> <li>• Przykrywa metody w klasach bazowych w celu dostosowania ich zachowania w klasach pochodnych.</li> <li>• Zrozumie, kiedy i dlaczego stosować polimorfizm.</li> <li>• Tworzy klasy abstrakcyjne i interfejsy, aby wprowadzać polimorfizm w programie.</li> </ul>	Test teoretyczny
Programuje wizualnie	<ul style="list-style-type: none"> <li>• Potrafi tworzyć aplikacje wykorzystujące graficzny interfejs użytkownika.</li> <li>• Korzysta z podstawowych komponentów takich jak etykieta, pole tekstowe, przyciski, pola wyboru, suwak.</li> <li>• Zapewnia interaktywność aplikacji, w tym obsługę myszy i klawiatury oraz rysowanie i formatowanie elementów interfejsu.</li> </ul>	Test teoretyczny
Wprowadza programowanie generyczne	<ul style="list-style-type: none"> <li>• Implementuje szablony funkcji i klas (lub klasy i metody generyczne) w swoim kodzie.</li> <li>• Rozumie różnicę między polimorfizmem statycznym a dynamicznym w kontekście programowania generycznego.</li> </ul>	Test teoretyczny

## Kwalifikacje

### Kompetencje

Usługa prowadzi do nabycia kompetencji.

#### Warunki uznania kompetencji

**Pytanie 1. Czy dokument potwierdzający uzyskanie kompetencji zawiera opis efektów uczenia się?**

Tak, dokument potwierdza opis efektów uczenia się.

**Pytanie 2. Czy dokument potwierdza, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji?**

Tak, dokument potwierdza, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji.

**Pytanie 3. Czy dokument potwierdza zastosowanie rozwiązań zapewniających rozdzielenie procesów kształcenia i szkolenia od walidacji?**

Tak, dokument potwierdza zastosowanie rozwiązań zapewniających rozdzielenie procesów kształcenia i szkolenia od walidacji.

# Program

## 1. Podstawowa terminologia związana z programowaniem obiektowym

- 1.1. Języki programowania obiektowego.
- 1.2. Pojęcia informatyczne związane z programowaniem obiektowym:  
klasa, obiekt, pola, metody, dziedziczenie.
- 1.3. Możliwości, które daje wykorzystanie programowania obiektowego.

## 2. Obsługa strumieni danych

- 2.1. Definicja strumienia danych, standardowe strumienie.
- 2.2. Strumienie znakowe i binarne.
- 2.3. Metody odczytu i zapisu danych.

## 3. Tworzenie i zapisywanie algorytmów

- 3.1. Tworzenie algorytmów rozwiązujących konkretne problemy programistyczne.
- 3.2. Zapisywanie algorytmów w formie pseudokodu i języka programowania.

## 4. Tworzenie klas. Obiekt – implementacja klasy

- 4.1. Tworzenie klasy: pola, metody, konstruktor.
- 4.2. Tworzenie obiektów: operator New.
- 4.3. Pola i metody statyczne.
- 4.4. Metody finalizujące (finalizery, destruktor).

## 5. Dziedziczenie

- 5.1. Zawieranie się klas a dziedziczenie
- 5.2. Przeciążanie metod (w tym konstruktorów).

## 6. Hermetyzacja

- 6.1. Przyczyny stosowania hermetyzacji
- 6.2. Modyfikatory hermetyzacji
- 6.3. Hermetyzacja a dziedziczenie

## 7. Polimorfizm

- 7.1. Przykrywanie metod
- 7.2. Przyczyny stosowania polimorfizmu
- 7.3. Metody wirtualne
- 7.4. Klasy abstrakcyjne (interfejsy)

## 8. Programowanie wizualne

- 8.1. Korzystanie z graficznego interfejsu użytkownika.
- 8.2. Wykorzystanie podstawowych komponentów: etykieta, pole tekstowe, przyciski, pola wyboru, suwak.
- 8.3. Interaktywność aplikacji. Obsługa myszy i klawiatury.
- 8.4. Rysowanie i wypełnianie obszarów, Tekst i czcionki.

## 9. Wprowadzenie do programowania generycznego

9.1. Szablony funkcji i klas (lub klasy i metody generyczne).

9.2. Polimorfizm statyczny a dynamiczny.

# Harmonogram

Liczba przedmiotów/zajęć: 16

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
<b>1 z 16</b> Podstawowa terminologia związana z programowaniem obiektowym.	-	16-09-2024	17:00	19:00	02:00
<b>2 z 16</b> Obsługa strumieni danych.	-	18-09-2024	16:00	19:00	03:00
<b>3 z 16</b> Tworzenie i zapisywanie algorytmów.	-	23-09-2024	16:30	19:00	02:30
<b>4 z 16</b> Tworzenie i zapisywanie algorytmów.	-	25-09-2024	16:30	19:00	02:30
<b>5 z 16</b> Tworzenie klas i implementacja obiektów.	-	30-09-2024	16:30	18:30	02:00
<b>6 z 16</b> Tworzenie klas i implementacja obiektów.	-	02-10-2024	16:30	18:00	01:30
<b>7 z 16</b> Dziedziczenie.	-	07-10-2024	16:30	19:00	02:30
<b>8 z 16</b> Dziedziczenie.	-	09-10-2024	16:30	19:00	02:30
<b>9 z 16</b> Hermetyzacja.	-	16-10-2024	16:30	19:00	02:30
<b>10 z 16</b> Hermetyzacja.	-	21-10-2024	16:30	19:00	02:30

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
<b>11 z 16</b> Polimorfizm.	-	21-10-2024	16:30	19:00	02:30
<b>12 z 16</b> Polimorfizm.	-	23-10-2024	16:30	19:00	02:30
<b>13 z 16</b> Programowanie wizualne.	-	28-10-2024	16:30	19:00	02:30
<b>14 z 16</b> Programowanie wizualne.	-	28-10-2024	16:30	19:00	02:30
<b>15 z 16</b> Programowanie generyczne.	-	30-10-2024	16:30	18:30	02:00
<b>16 z 16</b> Programowanie generyczne.	-	04-11-2024	16:30	17:45	01:15

## Cennik

### Cennik

Rodzaj ceny	Cena
Koszt usługi brutto	6 980,00 PLN
Koszt usługi netto	6 980,00 PLN
Koszt godziny brutto	139,60 PLN
Koszt godziny netto	139,60 PLN

## Prowadzący

Liczba prowadzących: 0

Brak wyników.

# Informacje dodatkowe

## Informacje o materiałach dla uczestników usługi

W ramach udziału w usłudze uczestnicy otrzymują materiały szkoleniowe: prezentacje oraz zestawy zadań.

## Informacje dodatkowe

**EGZAMIN WYMAGANY.** Każdy uczestnik musi przystąpić do egzaminu: ECCC z programowania na poziomie zaawansowanym.

Nazwa jednostki certyfikującej (egzaminującej): **Fundacja ECCC.**

Nazwa certyfikatu: **Europejski Certyfikat Kompetencji Informatycznych z programowania na poziomie zaawansowanym: CS M1 (ECCC).**

1 godzina usługi = 45 min lekcji dydaktycznej.

### Adres realizacji usługi:

**ul. Stanisława Staszica 30, 38-200 Jasło, woj. podkarpackie**

Usługodawca szkoleniowy posiada status **Adademii Edukacyjnej VCC.**

<https://vccsystem.eu/akademia-edukacyjna/>

## Adres

ul. Stanisława Staszica 30

38-200 Jasło

woj. podkarpackie

## Udogodnienia w miejscu realizacji usługi

- Wi-fi
- Laboratorium komputerowe
- Udogodnienia dla osób ze szczególnymi potrzebami

## Kontakt



**Damian Bartuś**

**E-mail** [biuro.proeducado@gmail.com](mailto:biuro.proeducado@gmail.com)

**Telefon** (+48) 665 558 011