

**OPIS MODUŁU KSZTAŁCENIA (przedmiot lub grupa przedmiotów)**

<b>Nazwa modułu/ przedmiotu</b> Programowanie		<b>Przedmiot/y</b> Programowanie I. Podstawy programowania Programowanie II. Programowanie obiektowe Programowanie III. Zaawansowane metody programowania Programowanie IV. Programowanie w języku Java Wprowadzenie do programowania aplikacji mobilnych Programowanie - projekt zespołowy			
<b>Nazwa jednostki prowadzącej przedmiot</b> Instytut Matematyki					
<b>kierunek</b>	<b>specjalność</b>	<b>specjalizacja</b>	<b>semestr/y</b>	<b>poziom kształcenia/ profil kształcenia</b>	<b>forma studiów</b>
Informatyka	Programowanie	-	1,2,3,4,5	SPS/ praktyczny	stacjonarne/niestacjonarne
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b> Dr Piotr Sulewski, dr inż. Zbigniew Ledóchowski, dr Ryszard Motyka					
<b>Formy zajęć</b>	<b>Liczba godzin</b>				<b>Liczba punktów ECTS</b>
	<b>N (nauczyciel)</b>		<b>S (student)</b>		
	<b>studia stacjonarne</b>	<b>studia niestacjonarne</b>	<b>studia stacjonarne</b>	<b>studia niestacjonarne</b>	
<b>Programowanie I. Podstawy programowania</b>	<b>60</b>	<b>36</b>	<b>60</b>	<b>84</b>	<b>4</b>
<b>(CL) Ćwiczenia laboratoryjne</b>	60	36	60	84	4
Przygotowanie do zajęć (w tym rozwiązywanie zadań domowych)			20	20	
Przygotowanie do kolokwiów			20	20	
Przygotowanie projektu			20	44	
<b>Programowanie II. Programowanie obiektowe</b>	<b>45</b>	<b>27</b>	<b>55</b>	<b>73</b>	<b>4</b>
<b>(CL) Ćwiczenia laboratoryjne</b>	45	27	55	73	4
Przygotowanie do zajęć (w tym rozwiązywanie zadań domowych)			15	23	
Przygotowanie do kolokwiów			20	30	
Przygotowanie projektu			20	20	
<b>Programowanie III. Zaawansowane metody programowania</b>	<b>45</b>	<b>27</b>	<b>45</b>	<b>63</b>	<b>3</b>
<b>(CL) Ćwiczenia laboratoryjne</b>	45	27	45	63	3
Przygotowanie do zajęć (w tym rozwiązywanie zadań domowych)			10	23	
Przygotowanie do kolokwiów			20	20	
Przygotowanie projektu			10	20	
<b>Programowanie IV. Programowanie w języku Java</b>	<b>45</b>	<b>27</b>	<b>55</b>	<b>73</b>	<b>4</b>
<b>(CL) Ćwiczenia laboratoryjne</b>	45	27	55	73	4
Przygotowanie do zajęć (w tym rozwiązywanie zadań domowych)			15	23	

Przygotowanie do kolokwiów			20	30	
Przygotowanie projektu			20	20	
<b>Wprowadzenie do programowania aplikacji mobilnych</b>	<b>45</b>	<b>27</b>	<b>80</b>	<b>98</b>	<b>5</b>
<b>(CL) Ćwiczenia laboratoryjne</b>	45	27	80	98	5
Przygotowanie do zajęć (w tym rozwiązywanie zadań domowych)			20	20	
Przygotowanie do kolokwiów			20	28	
Przygotowanie projektu			40	50	
<b>Programowanie - projekt zespołowy</b>	<b>15</b>	<b>9</b>	<b>60</b>	<b>66</b>	<b>3</b>
<b>(CL) Ćwiczenia laboratoryjne</b>	15	9	60	66	3
Przygotowanie do zajęć (w tym rozwiązywanie zadań domowych)			10	10	
Przygotowanie do kolokwiów			10	16	
Przygotowanie projektu			40	40	
<b>Razem</b>	<b>255</b>	<b>153</b>	<b>355</b>	<b>457</b>	<b>23</b>

#### Metody dydaktyczne

- (CL) ćwiczenia laboratoryjne: ćwiczenia praktyczne - rozwiązywanie zadań z wykorzystaniem komputera, metoda problemowa

#### Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

##### Programowanie I. Podstawy programowania

- Wymagania formalne: Wstęp do informatyki,
- Wymagania wstępne: wiadomości i umiejętności z przedmiotów informatycznych na poziomie szkoły ponadgimnazjalne

##### Programowanie II. Programowanie obiektowe

- Wymagania formalne: Wstęp do informatyki, Programowanie I. Podstawy programowania
- Wymagania wstępne: programowanie strukturalne

##### Programowanie III. Zaawansowane metody programowania

- Wymagania formalne: Programowanie I. Podstawy programowania, Programowanie II. Programowanie obiektowe
- Wymagania wstępne: umiejętność programowania strukturalnego i obiektowego w dowolnym języku

##### Programowanie IV. Programowanie w języku Java

- Wymagania formalne: Programowanie I. Podstawy programowania, Programowanie II. Programowanie obiektowe
- Wymagania wstępne: umiejętność programowania strukturalnego i obiektowego w dowolnym języku

##### Wprowadzenie do programowania aplikacji mobilnych

- Wymagania formalne: Programowanie I. Podstawy programowania, Programowanie II. Programowanie obiektowe
- Wymagania wstępne: umiejętność programowania strukturalnego i obiektowego w dowolnym języku

##### Programowanie - projekt zespołowy

- Wymagania formalne: wszystkie etapy programowania
- Wymagania wstępne: umiejętność programowania strukturalnego i obiektowego w dowolnym języku

#### Cel modułu

- Nauczenie rozwiązywania problemów programistycznych i pisanie programów w języku C#.
- Nauczenie zaawansowanych technik i mechanizmów programowania wykorzystywanych współcześnie, stosowanie ich w praktyce do implementacji poprawnych strukturalnie i efektywnych programów obiektowych.
- Nauczenie rozwiązywania problemów programistycznych i pisanie programów w języku Java
- Zdobyć praktycznych umiejętności programowania aplikacji mobilnych działających pod kontrolą najpopularniejszego systemu operacyjnego dedykowanego do urządzeń mobilnych tj. systemu Android
- Zdobyć wiedzy o specyfice, możliwościach i ograniczeniach aplikacji mobilnych działających pod kontrolą systemu Android.
- Nabycie umiejętności realizowania przedsięwzięć projektowych typu programistycznego (od założeń do programu)
- Nabycie umiejętności tworzenia prostej dokumentacji do realizowanych przedsięwzięć projektowych.

#### Programowanie I. Podstawy programowania

- Podstawowe koncepcje i środowisko C#
- Pierwsza aplikacja, kompilacja i uruchomienie programu
- Typy danych w C#
- Zmienne - deklaracje i przypisania

5. Wyprowadzanie danych na ekran
6. Operacje na zmiennych
7. Instrukcje warunkowa
8. Instrukcje wyboru
9. Pętle
10. Definicja procedury
11. Definicja funkcji
12. Przekazywanie parametrów i odbieranie wyników
13. Wzorcową strukturą programu
14. Podstawowe operacje na tablicach
15. Tablice wielowymiarowe
16. Ciągi znaków
17. Standardowe wejście i wyjście

#### **Programowanie II. Programowanie obiektowe**

1. Klasy i obiekty
2. Argumenty i przeciążanie metod
3. Konstruktory i destruktory
4. Klasy potomne
5. Modyfikatory dostępu
6. Przesłanie metod i składowe statyczne
7. Właściwości i struktury
8. Konwersje typów i rzutowanie obiektów
9. Późne wiązanie i wywoływanie metod klas pochodnych
10. Konstruktory oraz klasy abstrakcyjne
11. Tworzenie interfejsów
12. Implementacja kilku interfejsów
13. Klasa wewnątrz klasy
14. Kontrola typów i typy uogólnione

#### **Programowanie III. Zaawansowane metody programowania**

1. Operacje bazodanowe
2. Operacje na plikach
3. Operacje na systemie plików
4. Zapis i odczyt plików
5. Aplikacje z interfejsem graficznym
  - a. Tworzenie okien
  - b. Delegacje i zdarzenia
  - c. Komponenty graficzne
6. Alokacja pamięci
7. Obsługa wyjątków
8. Kompilacja i debugowanie - konfiguracja środowiska programisty
9. Programowanie w trybie konsoli np. w Linuxie
10. Testowanie programów

#### **Programowanie IV. Programowanie w języku Java**

1. Środowisko programistyczne Javy
2. Podstawowe elementy języka Java
3. Obiekty i klasy
4. Dziedziczenie
5. Interfejsy i klasy wewnętrzne
6. Obsługa zdarzeń
7. Komponenty Swing interfejsu użytkownika
8. Przygotowywanie apletów i aplikacji do użytku

#### **Wprowadzenie do programowania aplikacji mobilnych**

1. Wprowadzenie do zagadnienia natywnych aplikacji mobilnych, przegląd popularnych platform do rozwoju aplikacji mobilnych i zapoznanie z typami urządzeń klasyfikowanymi jako urządzenia mobilne.
2. Typowe narzędzia wspierające wytwarzanie natywnych aplikacji mobilnych, w tym środowisko programistyczne (IDE), emulator, tryb debugowania na urządzeniu. Przygotowywanie środowiska programistycznego. Organizacja plików w projekcie aplikacji.
3. Wstęp do programowania natywnej aplikacji mobilnej z uwzględnieniem programowania zdarzeniowego.
4. Programowanie interfejsów graficznych natywnych aplikacji mobilnych z uwzględnieniem specyficznych dla niej

standardów i zaleceń. Typowe kontrolki interfejsów graficznych użytkownika stosowane w aplikacjach mobilnych.

5. Nawigowanie w aplikacji.
6. Komunikacja wewnątrz aplikacji i między aplikacjami.
7. Stosowanie preferencji.
8. Obsługa multimediiów.
9. Przechowywanie danych (SQLite, odczyt i zapis do pliku, zapis w pamięci zewnętrznej).
10. RWD aplikacji mobilnych.
11. Przygotowanie aplikacji do dystrybucji i publikacji
12. Typowe biblioteki i interfejsy wejścia/wyjścia.

**Programowanie - projekt zespołowy**

1. Realizacja projektu programistycznego wykorzystującego poznane struktury danych, konstrukcje programistyczne oraz filozofie programowania, opartego na podanych wcześniej założeniach oraz uzupełnionego prostą dokumentacją (instrukcja użytkownika i opis programu).

Efekty kształcenia	Sposób zaliczenia oraz formy i podstawowe kryteria oceny/wymagania egzaminacyjne																								
<p><b>Wiedza</b></p> <p>W_01 zna proste, złożone i abstrakcyjne struktury danych oraz konstrukcje programistyczne stosowane w programowaniu</p> <p>W_02 zna narzędzia i techniki programowania obiektowego</p> <p>W_03 posiada szczegółową wiedzę na temat metod i języków programowania wykorzystywanych przy rozwiązywaniu zadań inżynierskich</p> <p>W_04 zna narzędzia i techniki programowania w Javie</p> <p>W_05 zna platformę programistyczną do tworzenia aplikacji mobilnych.</p> <p>W_06 ma wiedzę o specyfice, możliwościach i ograniczeniach aplikacji mobilnych.</p> <p>W_07 zna podstawowe kontrolki, konstrukcje programistyczne i zasady programowania aplikacji mobilnych.</p> <p>W_08 zna etapy realizowania przedsięwzięć projektowych typu programistycznego</p> <p><b>Umiejętności</b></p> <p>U_01 potrafi zastosować podstawowe typy i struktury danych oraz podstawowe konstrukcje programistyczne do konstrukcji prostych programów w języku programowania C#</p> <p>U_02 potrafi wydzielić logiczne i funkcjonalne fragmenty programu w postaci procedur (funkcji), potrafi stosować złożone struktury danych</p> <p>U_03 potrafi tworzyć obiektowe odwzorowanie dziedziny problemu wraz z realizacją związków między klasami pojęciowymi</p> <p>U_04 potrafi stosować w praktyce zaawansowane techniki programowania obiektowego</p> <p>U_05 potrafi korzystać z zaawansowanych technik programistycznych: dynamiczne zarządzanie pamięcią, dynamiczne struktury danych</p> <p>U_06 potrafi wychwytywać w dziedzinie problemu podstawowe wzorce projektowe, projektować ich wykorzystanie oraz je implementować.</p> <p>U_07 potrafi zastosować podstawowe typy i struktury danych oraz podstawowe konstrukcje programistyczne do konstrukcji prostych programów w języku programowania Java</p> <p>U_08 potrafi stosować w praktyce zaawansowane techniki programowania w Javie</p> <p>U_09 umie się posługiwać narzędziami do projektowania, tworzenia i debugowania oprogramowania działającego pod kontrolą systemu Android.</p> <p>U_10 umie projektować i wytwarzać proste aplikacje działających pod</p>	<p><b>A. Sposób zaliczenia</b></p> <p><b>Programowanie I. Podstawy programowania</b> CL – zaliczenie z oceną.</p> <p><b>Programowanie II. Programowanie obiektowe</b> CL – zaliczenie z oceną.</p> <p><b>Programowanie III. Zaawansowane metody programowania</b> CL – zaliczenie z oceną.</p> <p><b>Programowanie IV. Programowanie w języku Java</b> CL – zaliczenie z oceną.</p> <p><b>Wprowadzenie do programowania aplikacji mobilnych</b> CL – zaliczenie z oceną.</p> <p><b>Programowanie - projekt zespołowy</b> CL – zaliczenie z oceną.</p> <p><b>B. Sposoby weryfikacji i oceny efektów</b></p> <p><b>Programowanie I. Podstawy programowania</b> (CL) Ćwiczenia laboratoryjne kolokwium I – efekty: W_01, U_01 (50%) kolokwium II – efekty: W_01, U_02 (50%)</p> <p>Każda z form oceny CL jest punktowana, a suma punktów możliwych do uzyskania to y. Ocena dla zaliczenia CL jest ustalana na podstawie uzyskanej przez studenta sumy punktów P według zasady:</p> <table data-bbox="903 1178 1503 1368"> <tr> <td><math>P \in [0\% y, 50\% y)</math></td> <td>niedostateczna</td> </tr> <tr> <td><math>P \in [50\% y, 60\% y)</math></td> <td>dostateczna</td> </tr> <tr> <td><math>P \in [60\% y, 70\% y)</math></td> <td>dostateczna plus</td> </tr> <tr> <td><math>P \in [70\% y, 80\% y)</math></td> <td>dobra</td> </tr> <tr> <td><math>P \in [80\% y, 90\% y)</math></td> <td>db plus</td> </tr> <tr> <td><math>P \in [90\% y, 100\% y]</math></td> <td>bardzo dobra</td> </tr> </table> <p><b>Programowanie II. Programowanie obiektowe</b> (CL) Ćwiczenia laboratoryjne kolokwium I – efekty: W_02, U_03, U_04 (50%) kolokwium II – efekty: W_02, U_03, U_04 (50%)</p> <p>Każda z form oceny CL jest punktowana, a suma punktów możliwych do uzyskania to y. Ocena dla zaliczenia CL jest ustalana na podstawie uzyskanej przez studenta sumy punktów P według zasady:</p> <table data-bbox="903 1727 1503 1917"> <tr> <td><math>P \in [0\% y, 50\% y)</math></td> <td>niedostateczna</td> </tr> <tr> <td><math>P \in [50\% y, 60\% y)</math></td> <td>dostateczna</td> </tr> <tr> <td><math>P \in [60\% y, 70\% y)</math></td> <td>dostateczna plus</td> </tr> <tr> <td><math>P \in [70\% y, 80\% y)</math></td> <td>dobra</td> </tr> <tr> <td><math>P \in [80\% y, 90\% y)</math></td> <td>db plus</td> </tr> <tr> <td><math>P \in [90\% y, 100\% y]</math></td> <td>bardzo dobra</td> </tr> </table> <p><b>Programowanie III. Zaawansowane metody programowania</b></p>	$P \in [0\% y, 50\% y)$	niedostateczna	$P \in [50\% y, 60\% y)$	dostateczna	$P \in [60\% y, 70\% y)$	dostateczna plus	$P \in [70\% y, 80\% y)$	dobra	$P \in [80\% y, 90\% y)$	db plus	$P \in [90\% y, 100\% y]$	bardzo dobra	$P \in [0\% y, 50\% y)$	niedostateczna	$P \in [50\% y, 60\% y)$	dostateczna	$P \in [60\% y, 70\% y)$	dostateczna plus	$P \in [70\% y, 80\% y)$	dobra	$P \in [80\% y, 90\% y)$	db plus	$P \in [90\% y, 100\% y]$	bardzo dobra
$P \in [0\% y, 50\% y)$	niedostateczna																								
$P \in [50\% y, 60\% y)$	dostateczna																								
$P \in [60\% y, 70\% y)$	dostateczna plus																								
$P \in [70\% y, 80\% y)$	dobra																								
$P \in [80\% y, 90\% y)$	db plus																								
$P \in [90\% y, 100\% y]$	bardzo dobra																								
$P \in [0\% y, 50\% y)$	niedostateczna																								
$P \in [50\% y, 60\% y)$	dostateczna																								
$P \in [60\% y, 70\% y)$	dostateczna plus																								
$P \in [70\% y, 80\% y)$	dobra																								
$P \in [80\% y, 90\% y)$	db plus																								
$P \in [90\% y, 100\% y]$	bardzo dobra																								

kontrolą systemu Android.

U\_11 realizuje zespołowe projekty programistyczne, wypełniając w zespole projektowym różne role

### Kompetencje społeczne

K\_01 zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego kształcenia, jest otwarty na poszukiwanie rozwiązań.

(CL) Ćwiczenia laboratoryjne

kolokwium I – efekty: W\_03, U\_05, U\_06 (50%)

kolokwium II – efekty: W\_03, U\_05, U\_06 (50%)

Każda z form oceny CL jest punktowana, a suma punktów możliwych do uzyskania to y.

Ocena dla zaliczenia CL jest ustalana na podstawie uzyskanej przez studenta sumy punktów P według zasady:

$P \in [0\% y, 50\% y)$	niedostateczna
$P \in [50\% y, 60\% y)$	dostateczna
$P \in [60\% y, 70\% y)$	dostateczna plus
$P \in [70\% y, 80\% y)$	dobra
$P \in [80\% y, 90\% y)$	db plus
$P \in [90\% y, 100\% y]$	bardzo dobra

### Programowanie IV. Programowanie w języku Java

(CL) Ćwiczenia laboratoryjne

kolokwium I – efekty: W\_04, U\_07, U\_08 (50%)

kolokwium II – efekty: W\_04, U\_07, U\_08 (50%)

Każda z form oceny CL jest punktowana, a suma punktów możliwych do uzyskania to y.

Ocena dla zaliczenia CL jest ustalana na podstawie uzyskanej przez studenta sumy punktów P według zasady:

$P \in [0\% y, 50\% y)$	niedostateczna
$P \in [50\% y, 60\% y)$	dostateczna
$P \in [60\% y, 70\% y)$	dostateczna plus
$P \in [70\% y, 80\% y)$	dobra
$P \in [80\% y, 90\% y)$	db plus
$P \in [90\% y, 100\% y]$	bardzo dobra

### Wprowadzenie do programowania aplikacji mobilnych

(CL) Ćwiczenia laboratoryjne

- kolokwium pisemne o charakterze praktycznym (rozwiązywanie zadań praktycznych) – efekty: W\_05, W\_06, W\_07, U\_09, U\_10, (50%)

- domowa praca kontrolna (przygotowanie prostej aplikacji mobilnej) - efekty: W\_05, W\_06, W\_07, U\_09, U\_10, K\_01 (50%)

Każda z form oceny CL jest punktowana, a suma punktów możliwych do uzyskania to y.

W nawiasach przy każdej z form oceniania CL podano jaki procent y można uzyskać maksymalnie z tej formy oceniania. Ocena A dla zaliczenia CL jest ustalana na podstawie uzyskanej przez studenta sumy punktów P według zasady:

$P \in [0\% y, 50\% y)$	niedostateczna
$P \in [50\% y, 60\% y)$	dostateczna
$P \in [60\% y, 70\% y)$	dostateczna plus
$P \in [70\% y, 80\% y)$	dobra
$P \in [80\% y, 90\% y)$	db plus
$P \in [90\% y, 100\% y]$	bardzo dobra

Końcowa ocena z zaliczenia przedmiotu jest tożsama z

	<p>oceną A.</p> <p><b>Programowanie - projekt zespołowy</b> (CL) Ćwiczenia laboratoryjne Przygotowanie projektu – efekt W_08, U_11. Suma punktów możliwych do uzyskania za projekt to y. Ocena dla zaliczenia CL jest ustalana na podstawie uzyskanej przez studenta sumy punktów P według zasady:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 70%;">P ∈ [0% y, 50% y)</td> <td>niedostateczna</td> </tr> <tr> <td>P ∈ [50% y, 60% y)</td> <td>dostateczna</td> </tr> <tr> <td>P ∈ [60% y, 70% y)</td> <td>dostateczna plus</td> </tr> <tr> <td>P ∈ [70% y, 80% y)</td> <td>dobra</td> </tr> <tr> <td>P ∈ [80% y, 90% y)</td> <td>db plus</td> </tr> <tr> <td>P ∈ [90% y, 100% y]</td> <td>bardzo dobra</td> </tr> </table>	P ∈ [0% y, 50% y)	niedostateczna	P ∈ [50% y, 60% y)	dostateczna	P ∈ [60% y, 70% y)	dostateczna plus	P ∈ [70% y, 80% y)	dobra	P ∈ [80% y, 90% y)	db plus	P ∈ [90% y, 100% y]	bardzo dobra
P ∈ [0% y, 50% y)	niedostateczna												
P ∈ [50% y, 60% y)	dostateczna												
P ∈ [60% y, 70% y)	dostateczna plus												
P ∈ [70% y, 80% y)	dobra												
P ∈ [80% y, 90% y)	db plus												
P ∈ [90% y, 100% y]	bardzo dobra												

**Matryca efektów kształcenia dla przedmiotu**

Numer (symbol) efektu kształcenia	Odniesienie do efektów kształcenia dla programu	Odniesienie do charakterystyki drugiego stopnia PRK dla obszaru/ obszarów
W_01	K1_W20, K1_W21	P6S_WG
W_02	K1_W20, K1_W21	P6S_WG
W_03	K1_W19, K1_W21	P6S_WG
W_04	K1_W19, K1_W21	P6S_WG
W_05	K1_W21	P6S_WG
W_06	K1_W03, K1_W19	P6S_WG
W_07	K1_W19, K1_W22	P6S_WG
W_08	K1_W19, K1_W20	P6S_WG
W_09	K1_W19, K1_W23	P6S_WG
U_01	K1_U28	P6S_UW
U_02	K1_U28	P6S_UW
U_03	K1_U28, K1_U29	P6S_UW
U_04	K1_U28, K1_U29	P6S_UW
U_05	K1_U28, K1_U29, K1_U32	P6S_UW
U_06	K1_U28, K1_U29	P6S_UW
U_07	K1_U28, K1_U29, K1_U31	P6S_UW
U_08	K1_U28, K1_U29, K1_U31	P6S_UW
U_09	K1_U28, K1_U29	P6S_UW
U_10	K1_U35	P6S_UW
U_11	K1_U36	P6S_UW
K_01	K1_K01, K1_K02	P6S_KK, P6S_KO

**A. Wykaz literatury**

- a. J. Hilyard , S. Teilhet. C# Księga przepisów. APN Promise 2016.
- b. J. Albahari, B. Albahari. C# 5.0. Leksykon kieszonkowy. Helion 2013.
- c. J. Albahari, B. Albahari. C# 6.0 w pigułce Helion 2016.

- d. Delessio C., Darcey L., Conder S., *Android Studio w 24 godziny. Wygodne programowanie dla platformy Android. Wydanie IV*, Helion 2016
  - e. Darwin F. I., *Android. Receptury*, Helion 2013
  - f. Kurs Video *Android Development For Absolute Beginners* dostępny pod adresem <https://www.youtube.com/playlist?list=PLB03EA9545DD188C3>
  - g. Kurs Video *Anroid Tutorial 2015* dostępny pod adresem <https://www.youtube.com/playlist?list=PLshdtb5UWjSp0879mLeCsDQN6L73XBZTk>
  - h. Google Inc., <https://developer.android.com/guide/index.html>
  - i. C. S. Horstmann. *Java. Podstawy*. Wydanie X. Helion.
  - j. M. Lis. *Java. Praktyczny kurs*. Wydanie IV.
  - k. S. Cleary. *Współbieżność w języku C#. Receptury*
  - l. M. Kubiak. *C#. Zadania z programowania z przykładowymi rozwiązaniami (ebook)*
  - m. J. Matulewski. *Visual Studio 2013. Podręcznik programowania w C# z zadaniami*
- B. Literatura uzupełniająca
- a. J. Skeet. *C# od podszewki*. Helion 2012.
  - b. I. Griffiths, M. Adams, J. Liberty. *C# Programowanie*. Helion 2012.
  - c. J. Matulewski. *Wzorce projektowe dla programistów C#. Kurs video*. Helion Videopoint
  - d. Collins Ch., Galpin M., Kaeppler M., *Android w praktyce*, Helion 2012
  - e. Lehtimaki J., *Android UI. Podręcznik dla projektantów. Smashing Magazine*, Helion 2013
  - f. Annuzzi J. Jr., Darcey L., Conder S., *Android. Wprowadzenie do programowania aplikacji. Wydanie V*, Helion 2016
  - g. Stasiewicz A., *Android Studio. Podstawy tworzenia aplikacji*, Helion 2015
  - h. K. Sierra, B. Bates. *Java. Rusz głową – Wydanie II*.
  - i. J. Roszkowski. *Analiza i projektowanie strukturalne. Wydanie III*

**Kontakt**

dr Piotr Sulewski

[piotr.sulewski@apsl.edu.pl](mailto:piotr.sulewski@apsl.edu.pl)