

KARTA PRZEDMIOTU DLA NABORU 2022/2023 FORMA STUDIÓW: STACJONARNA					
INFORMACJE OGÓLNE					
1. Nazwa przedmiotu Algorytmy i struktury danych					
2. Nazwa kierunku Informatyka					
3. Poziom kształcenia Studia pierwszego stopnia					
4. Liczba punktów ECTS 5					
5. Liczba godzin w semestrze					
semestr	w	ćw	lab/lek	prj/zp	prk
III	30		30		
6. Język wykładowy polski					
7. Wykładowca dr inż. Marcin Klimek					
INFORMACJE SZCZEGÓŁOWE					
8. Wymagania wstępne					
1. Znajomość elementów matematyki dyskretnej tj. teoria grafów itp.					
2. Wiedza z podstaw analizy matematycznej.					
3. Umiejętność programowania w zakresie podstawowym.					
9. Cele przedmiotu					
C1. Zapoznanie z podstawowymi strukturami danych i możliwościami ich wykorzystania w programach komputerowych.					
C2. Zdobycie wiedzy dotyczącej technik algorytmicznych i możliwości ich zastosowania w różnych problemach optymalizacyjnych.					
C3. Zaznajomienie z podstawowymi algorytmami dla ważnych zagadnień informatycznych tj. sortowanie, wyszukiwanie danych itp.					
C4. Zdobycie umiejętności zapisu procedur obliczeniowych w postaci pseudokodu, schematów blokowych.					
C5. Zrozumienie zasad oceny przydatności algorytmu w konkretnym zastosowaniu m.in. ze względu na złożoność obliczeniową i pamięciową.					
10. Efekty uczenia się w zakresie wiedzy, umiejętności i kompetencji społecznych					
Student, który zaliczył przedmiot:				odniesienie do kierunkowych efektów uczenia się	
WIEDZA					
EU01	Zna i rozumie pojęcia dotyczące implementacji wydajnych algorytmów dla praktycznych problemów obliczeniowych			K_W01, K_W07	
EU02	Zna i rozumie pojęcia dotyczące złożoności obliczeniowej i doboru odpowiednich technik algorytmicznych			K_W01, K_W07	
UMIEJĘTNOŚCI					
EU03	Potrafi implementować wydajne algorytmy, stosować odpowiednie techniki algorytmiczne dla praktycznych problemów obliczeniowych			K_U06, K_U08, K_U22	

EU04	Potrafi wybierać wydajne struktury danych przy konstruowaniu programów komputerowych	K_U06, K_U22
EU05	Potrafi oceniać złożoność obliczeniową zastosowanych procedur obliczeniowych i identyfikować trudne zagadnienia obliczeniowe, w których uzasadnione jest wykorzystanie rozwiązań przybliżonych	K_U06, K_U22
EU06	Potrafi zapisywać rozwiązania, algorytmy w postaci pseudokodu i schematów blokowych	K_U08
KOMPETENCJE SPOŁECZNE		
EU07	Jest gotów do uznania potrzeby poznawania nowych algorytmów	K_K01
EU08	Jest gotów do odpowiedzialnego pełnienia ról zawodowych	K_K04
11. Treści programowe		
Forma zajęć – wykłady/ ćwiczenia/laboratoria/zajęcia praktyczne itp.		
Wykład <ol style="list-style-type: none"> 1. Wprowadzenie do algorytmiki. Definicje algorytmu. Cechy algorytmu. Sposoby zapisu algorytmów. Liczby pseudolosowe. 2. Złożoność obliczeniowa. Problemy NP-trudne i NP-zupełne. 3. Metody tworzenia algorytmów: dziel i zwyciężaj, metoda zachłanna, programowanie dynamiczne, rekurencja, heurystyki itd. 4. Przykładowe problemy optymalizacyjne i wybrane procedury ich rozwiązywania (problem plecakowy, wydawania reszty, komiwojażera, wieża Hanoi itp.). 5. Algorytmy sortowania (szybkie, przez wstawianie, wybieranie, bąbelkowe, pozycyjne, kopcowe) 6. Analiza złożoności wybranych algorytmów (np. sortowania). 7. Typy danych: tablice, stosy, kolejki, listy jedno- i dwukierunkowe. Podstawowe operacje na 8. Struktury danych w C#. Efektywne wyszukiwanie (binarne), haszowanie. Słowniki. 9. Grafy. Cykl Eulera, Hamiltona. Algorytm Dijkstry. 10. Drzewa. BST, AVL, B drzewa. Kopiec. Podstawowe operacje na drzewach Laboratorium <ol style="list-style-type: none"> 1. Implementacja w C# prostych algorytmów np. znajdowanie liczb pierwszych, szukanie minimum funkcji, operacje na macierzach itp. 2. Kodowanie w C# podstawowych operacji na listach, tablicach tj. wyszukiwanie, dodawanie, usuwanie elementów. 3. Konstruowanie algorytmów rekurencyjnych i iteracyjnych (wyszukiwanie binarne, obliczanie silni, ciąg Fibonacciego, wieża Hanoi). 4. Budowa w C# algorytmów zachłannych i siłowych dla przykładowych problemów tj. problem wydawania reszty, plecakowy, komiwojażera itp. 5. Budowa klasycznych algorytmów sortowania : bąbelkowy, przez wstawianie, wybieranie, szybkie sortowanie. Analiza złożoności obliczeniowej. 6. Budowa dynamicznych struktur danych: stosów, kolejek, list i podstawowych operacji dla tych struktur. Korzystanie z kolekcji w C#. 7. Implementacja w C# algorytmów przeszukiwania drzew, grafów, algorytmu Dijkstry. 		
12. Narzędzia/metody dydaktyczne		
<ol style="list-style-type: none"> 1. Podczas laboratoriów praca indywidualna studentów: pisanie programów komputerowych, rozwiązywanie problemów algorytmicznych, analiza kodu programów komputerowych pod kątem wydajności, optymalności. 2. Laboratoria w części odbywają się przy tablicy: studenci rozwiązują zagadnienia teoretyczne tj. sortowanie liczb, obliczanie złożoności obliczeniowej. 3. Wykład prowadzony z wykorzystaniem projektora multimedialnego. Wykłady przygotowane w formie prezentacji PowerPoint. 4. Podczas wykładów prezentowane są także wybrane programy komputerowe w C# tj. generowanie liczb pseudolosowych (Lotto), wyszukiwanie binarne, przeglądanie drzew, sortowanie szybkie, przez 		

wstawianie, implementacje stosu, kolejki, listy dwukierunkowej, przykłady obsługi kolekcji Stack, Queue, Dictionary, SortedList, algorytm Dijkstry, generowanie wszystkich permutacji n liczb itp.	
5. Podczas laboratoriów praca indywidualna studentów: pisanie programów komputerowych, rozwiązywanie problemów algorytmicznych, analiza kodu programów komputerowych pod kątem wydajności, optymalności.	
6. Dyskusja	
7. Konsultacje	
13. Sposoby oceny (częstkowe, końcowe)	
1. Dwa kolokwia przy komputerze: implementacja programów stosujących różne struktury danych i techniki algorytmiczne.	
2. Kolokwium pisemne ze złożoności obliczeniowej algorytmów.	
3. Ocena realizacji projektu zaliczeniowego do zrealizowania w domu (implementacja wybranego algorytmu rozwiązującego problem optymalizacyjny).	
4. Kolokwium pisemne z algorytmów sortowania.	
5. Ocena ciągła pracy indywidualnej (realizacji programów komputerowych) podczas laboratoriów.	
6. Egzamin w formie pisemnej sprawdzający wiedzę z wykładów i laboratoriów.	
14. Obciążenie pracą studenta	
Forma aktywności	liczba godzin
1. Zajęcia z bezpośrednim udziałem nauczyciela oraz konsultacje	70
2. Nakład pracy studenta	55
suma	125
liczba punktów ECTS	5
15. Literatura	
Literatura podstawowa:	
1. Wirth N., Algorytmy + struktury danych = programy. WNT 2004.	
2. Cormen Thomas H. Algorytmy bez tajemnic, Gliwice : Helion, 2013.	
3. Wróblewski P., Algorytmy, struktury danych i techniki programowania. Helion 2010.	
4. Lafore R., Java. Algorytmy i struktury danych. Helion 2004.	
5. Knuth D. Sztuka programowania. Tom 1. Algorytmy podstawowe, Tom 2. Algorytmy suminumeryczne. Warszawa: WNT 2003.	
Literatura uzupełniająca:	
1. Banachowski L., Diks K., Rytter W., Algorytmy i struktury danych. WNT 1996.	
2. Cormen T. H., Leiserson C. E., Rivest R. L., Stein C., Wprowadzenie do algorytmów. WNT Warszawa 2004.	
3. Stephens R., Algorytmy i Struktury Danych. Helion Gliwice 2000.	
4. Dasgupta S., Papadimitriou Ch., Vazirani U.; Algorytmy, Warszawa : Wydawnictwo Naukowe PWN, 2010.	
16. Formy oceny – szczegóły	
Warunki uzyskania zaliczenia przedmiotu: zajęcia kończą się egzaminem, do którego można przystąpić po zaliczeniu laboratoriów. Egzamin jest przeprowadzany w formie pisemnej (pytania otwarte i testowe), jego zaliczenie jest możliwe po uzyskaniu minimum 51% punktów.	
<u>Sposób weryfikacji efektów uczenia się:</u>	
Ocena stopnia osiągniętych przez studenta efektów uczenia się następuje wg poniższych kryteriów:	
5.0 – zakładany efekt uczenia się został osiągnięty bez zastrzeżeń	
4.5 – zakładany efekt uczenia się został osiągnięty z pojedynczymi brakami/błędami	
4.0 – zakładany efekt uczenia się został osiągnięty z nielicznymi brakami/błędami	
3.5 – zakładany efekt uczenia się został osiągnięty z wieloma brakami/błędami	
3.0 – zakładany efekt kształcenia został osiągnięty z licznymi i istotnymi brakami/błędami (minimalnie wymagany poziom osiągnięcia efektu)	
2.0 – zakładany efekt uczenia się nie został osiągnięty	
17. Inne przydatne informacje o przedmiocie	

1. Bezpośrednich informacji o problematyce zajęć i treściach programowych udziela Prowadzący w trakcie zajęć i podczas konsultacji
2. Zajęcia odbywać się będą w Akademii Bialskiej im. Jana Pawła II
3. Zajęcia odbywać się będą zgodnie z aktualnym planem zajęć
4. Konsultacje odbywać się będą zgodnie z obowiązującym terminarzem