



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez  
Unię Europejską w ramach  
Europejskiego Funduszu  
Społecznego

**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



<b>Nazwa przedmiotu</b>		<b>Kod ECTS</b>	
Algorytmy kombinatoryczne		11.3.0459	
<b>Nazwa jednostki prowadzącej przedmiot</b>			
Instytut Informatyki			
<b>Studia</b>			
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>drugiego stopnia</b>
Wydział Matematyki, Fizyki i Informatyki	Informatyka	<b>forma</b>	niestacjonarne (zaoczne)
		<b>moduł</b>	wszystkie
		<b>specjalnościowy</b>	wszystkie
		<b>specjalizacja</b>	wszystkie
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>			
prof. UG, dr hab. Tomasz Dzido			
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>	
<b>Formy zajęć</b>		6 Przedmiot w wymiarze 30h wykładu i 30h ćwiczeń/lab. + praca własna studenta.	
Wykład, Ćw. laboratoryjne			
<b>Sposób realizacji zajęć</b>			
zajęcia w sali dydaktycznej			
<b>Liczba godzin</b>			
Wykład: 20 godz., Ćw. laboratoryjne: 20 godz.			
<b>Cykl dydaktyczny</b>			
2017/2018 letni			
<b>Status przedmiotu</b>		<b>Język wykładowy</b>	
fakultatywny (do wyboru)		polski	
<b>Metody dydaktyczne</b>		<b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>	
<ul style="list-style-type: none"> <li>- Projektowanie doświadczeń</li> <li>- Rozwiązywanie zadań</li> <li>- Wykonywanie doświadczeń</li> <li>- Wykład z prezentacją multimedialną</li> </ul>		<b>Sposób zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Egzamin</li> </ul>	
		<b>Formy zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- egzamin pisemny z pytaniami (zadaniami) otwartymi</li> <li>- ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru</li> <li>- kolokwium</li> </ul>	
		<b>Podstawowe kryteria oceny</b>	
		Przedmiot kończy się egzaminem pisemnym, na zaliczenie potrzeba zdobyć 50% punktów. Aby do niego podejść należy najpierw zaliczyć ćwiczenia, na których będzie kolokwium z zadań - również 50% zdobytych punktów daje ich zaliczenie. Dodatkowe punkty na ćwiczeniach można uzyskać za implementację niektórych algorytmów.	
<b>Sposób weryfikacji założonych efektów kształcenia</b>			
<b>Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi</b>			
<b>A. Wymagania formalne</b>			
Wcześniejsze ukończenie przedmiotu Matematyka Dyskretna.			
<b>B. Wymagania wstępne</b>			
Wymagana jest podstawowa znajomość pojęć kombinatorycznych, tak jak jest to wykładane na przedmiocie Matematyka Dyskretna.			
<b>Cele kształcenia</b>			
Celem przedmiotu jest zapoznanie studentów z budową, sposobami możliwej implementacji, własnościami i zastosowaniem licznych algorytmów			

<p>kombinatorycznych.</p>	
<p><b>Treści programowe</b></p> <p>Na wykładzie poruszone zostaną takie zagadnienia jak:</p> <ol style="list-style-type: none"> <li>1. Reprezentacje obiektów kombinatorycznych</li> <li>2. Generowanie obiektów kombinatorycznych</li> <li>3. Własności permutacji i ich generowanie</li> <li>4. Generowanie kombinacji, metoda minimalnych zmian</li> <li>5. Kompozycje i rozkłady liczb</li> <li>6. Wybrane algorytmy teoriografowe</li> <li>7. Lemat Burnside'a i twierdzenie Polyi</li> <li>8. Metody przeszukiwania przestrzeni rozwiązań</li> <li>9. Algorytmy zachłanne</li> <li>10. Algorytmy rekurencyjne</li> <li>11. Metoda podziału i ograniczeń</li> <li>12. Programowanie dynamiczne</li> <li>13. Algorytm KMP oraz Boyera-Moore'a znajdowania łańcucha znaków w tekście</li> <li>14. Drzewa sufiksowe a grafy podstów</li> </ol>	
<p><b>Wykaz literatury</b></p> <p>A. V. Aho, J. E. Hopcroft, J. D. Ullman: Projektowanie i analiza algorytmów. Helion, 2003  W. Lipski: Kombinatoryka dla programistów, WNT 2007  T. H. Cormen, Ch. E. Leiserson, R. L. Rivest: Wprowadzenie do algorytmów. WNT, 1998.  E.M.Reingold, J.Nievergelt, N.Deo: Algorytmy kombinatoryczne. PWN, 1985.</p>	
<p><b>Efekty kształcenia (obszarowe i kierunkowe)</b></p> <p>K_W02 ma pogłębioną wiedzę na temat podstawowych paradygmatów programowania; zna również aktualne trendy w językach programowania  K_W04 zna formalne modele obliczeń a także ich własności i znaczenie w praktycznych zastosowaniach informatycznych, ma wiedzę na temat barier obliczalności i trudności obliczeń  K_W05 Zna najważniejsze konstrukcje programistyczne oraz struktury danych  K_W06 zna zaawansowane metody projektowania i analizowania złożoności obliczeniowej algorytmów, zna zasady działania oraz praktycznego zastosowania najważniejszych algorytmów różnego typu w sensie ich treści jak i sposobu ich wykonywania  K_W09 zna biegle co najmniej dwa języki programowania oraz biblioteki algorytmów i struktur danych oraz ma wiedzę na temat praktycznych uwarunkowań wydajnych implementacji algorytmów  K_W10 zna metody algorytmicznego rozwiązywania problemów obliczeniowo trudnych (aproxymacja, szybkie algorytmy wykładnicze, heurystyki)  K_W12 zna dobrze zasady bezpieczeństwa i higieny pracy w zawodzie informatyka  K_U02 ma umiejętność projektowania abstrakcyjnych struktur danych i ich wydajnych implementacji  K_U06 projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy z wykorzystaniem różnych technik programistycznych  K_U07 potrafi zastosować znane algorytmy w konkretnych sytuacjach, potrafi efektywnie dobrać rodzaj i sposób wykonania algorytmu w zależności od postawionego problemu</p>	<p><b>Wiedza</b></p> <p>Student:</p> <ul style="list-style-type: none"> <li>• ma pogłębioną wiedzę na temat podstawowych paradygmatów programowania potrzebnych do implementacji algorytmów</li> <li>• ma wiedzę na temat barier obliczalności i trudności obliczeń złożonych algorytmów</li> <li>• zna najważniejsze konstrukcje programistyczne oraz struktury danych</li> <li>• zna zaawansowane metody projektowania algorytmów, zna zasady działania oraz praktycznego zastosowania najważniejszych algorytmów różnego typu w sensie ich treści jak i sposobu ich wykonywania</li> <li>• zna biegle co najmniej dwa języki programowania oraz biblioteki algorytmów i struktur danych</li> <li>• zna różne metody algorytmicznego rozwiązywania problemów obliczeniowo trudnych</li> <li>• zna dobrze zasady bezpieczeństwa i higieny pracy w zawodzie informatyka</li> </ul> <p><b>Umiejętności</b></p> <p>Student:</p> <ul style="list-style-type: none"> <li>• ma umiejętność projektowania abstrakcyjnych struktur danych i ich wydajnych implementacji</li> <li>• projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy z wykorzystaniem różnych technik programistycznych</li> <li>• potrafi zastosować znane algorytmy w konkretnych sytuacjach, potrafi efektywnie dobrać rodzaj i sposób wykonania algorytmu w zależności od postawionego problemu</li> </ul> <p><b>Kompetencje społeczne (postawy)</b></p>
<p><b>Kontakt</b></p> <p>tdz@inf.ug.edu.pl</p>	