

**AGH**AGH UNIVERSITY OF SCIENCE
AND TECHNOLOGY

Nazwa modułu zajęć:	Programowanie Dyskretne				
Rok akademicki:	2019/2020	Kod:	AMAT-2-301-MZ-s	Punkty ECTS:	6
Wydział:	Matematyki Stosowanej				
Kierunek:	Matematyka	Specjalność:	Matematyka w zarządzaniu		
Poziom studiów:	Studia II stopnia	Forma studiów:	Stacjonarne		
Język wykładowy:	Polski	Profil:	Ogólnoakademicki (A)	Semestr:	3
Strona www:	—				
Prowadzący moduł:	dr hab. Żak Andrzej (zakandr@agh.edu.pl)				

Treści programowe zapewniające uzyskanie efektów uczenia się dla modułu zajęć

Problemy programowania dyskretnego (załadunku, przydziału, komiwojażera, lokalizacyjno-transportowy, pokrycia zbioru itd.) oraz ich modele matematyczne w postaci problemów progr. całkowitoliczbowego.

Opis efektów uczenia się dla modułu zajęć

Kod MEU	Student, który zaliczył moduł zajęć zna i rozumie/potrafi/jest gotów do	Powiązania z KEU	Sposób weryfikacji i oceny efektów uczenia się osiągniętych przez studenta w ramach poszczególnych form zajęć i dla całego modułu zajęć
Wiedza: zna i rozumie			
M_W001	zna podstawowe problemy programowania dyskretnego (załadunku, przydziału, komiwojażera, lokalizacyjno-transportowy, pokrycia zbioru itd.) oraz ich modele matematyczne w postaci problemów programowania całkowitoliczbowego.	MAT2A_W05, MAT2A_U02, MAT2A_K05, MAT2A_U16, MAT2A_W04	Aktywność na zajęciach, Egzamin, Kolokwium, Odpowiedź ustna
M_W002	zna podstawowe metody rozwiązywania problemów programowania całkowitoliczbowego i binarnego.	MAT2A_W05, MAT2A_U14, MAT2A_W04	Aktywność na zajęciach, Egzamin, Kolokwium, Odpowiedź ustna
Umiejętności: potrafi			

M_U001	potrafi samodzielnie zbudować model matematyczny (model problemu programowania całkowitoliczbowego) dla różnorodnych zagadnień praktycznych oraz znaleźć jego rozwiązanie za pomocą dostępnych programów komputerowych.	MAT2A_W05, MAT2A_U16, MAT2A_W04	Aktywność na zajęciach, Egzamin, Kolokwium, Odpowiedź ustna
M_U002	potrafi samodzielnie przeprowadzić proste dowody wykorzystując poznaną wiedzę z programowania dyskretnego	MAT2A_U03, MAT2A_K01, MAT2A_U02, MAT2A_U14, MAT2A_U01, MAT2A_W02, MAT2A_K02	Aktywność na zajęciach, Egzamin, Kolokwium, Odpowiedź ustna
M_U003	potrafi wykorzystać wiedzę z innych działów matematyki (algebra, matematyka dyskretna, programowanie liniowe, teoria grafów) w programowaniu dyskretnym	MAT2A_U14, MAT2A_W07, MAT2A_U04	Aktywność na zajęciach, Egzamin, Kolokwium, Odpowiedź ustna
Kompetencje społeczne: jest gotów do			
M_K001	umie ocenić stopień zrozumienia przez siebie problemu i brakujące elementy rozumowania	MAT2A_K01, MAT2A_K02	Aktywność na zajęciach, Egzamin, Kolokwium, Odpowiedź ustna

Liczba godzin zajęć w ramach poszczególnych form zajęć

Suma	Forma zajęć dydaktycznych										
	Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Prace kontrolne i przejściowe	Lektorat
60	30	0	30	0	0	0	0	0	0	0	0

Matryca kierunkowych efektów uczenia się w odniesieniu do form zajęć i sposobu zaliczenia, które pozwalają na ich uzyskanie

Kod MEU	Student, który zaliczył moduł zajęć zna i rozumie/potrafi/jest gotów do	Forma zajęć dydaktycznych										
		Wykład	Ćwiczenia audytoryjne	Ćwiczenia laboratoryjne	Ćwiczenia projektowe	Konwersatorium	Zajęcia seminaryjne	Zajęcia praktyczne	Zajęcia terenowe	Zajęcia warsztatowe	Prace kontrolne i przejściowe	Lektorat
Wiedza: zna i rozumie												

M_W001	zna podstawowe problemy programowania dyskretnego (załadunku, przydziału, komiwojażera, lokalizacyjno-transportowy, pokrycia zbioru itd.) oraz ich modele matematyczne w postaci problemów programowania całkowitoliczbowego.	+	-	+	-	-	-	-	-	-	-	-
M_W002	zna podstawowe metody rozwiązywania problemów programowania całkowitoliczbowego i binarnego.	+	-	+	-	-	-	-	-	-	-	-
Umiejętności: potrafi												
M_U001	potrafi samodzielnie zbudować model matematyczny (model problemu programowania całkowitoliczbowego) dla różnorodnych zagadnień praktycznych oraz znaleźć jego rozwiązanie za pomocą dostępnych programów komputerowych.	+	-	+	-	-	-	-	-	-	-	-
M_U002	potrafi samodzielnie przeprowadzić proste dowody wykorzystując poznaną wiedzę z programowania dyskretnego	+	-	+	-	-	-	-	-	-	-	-
M_U003	potrafi wykorzystać wiedzę z innych działów matematyki (algebra, matematyka dyskretna, programowanie liniowe, teoria grafów) w programowaniu dyskretnym	+	-	+	-	-	-	-	-	-	-	-
Kompetencje społeczne: jest gotów do												
M_K001	umie ocenić stopień zrozumienia przez siebie problemu i brakujące elementy rozumowania	+	-	+	-	-	-	-	-	-	-	-

Nakład pracy studenta (bilans punktów ECTS)

Forma aktywności studenta	Obciążenie studenta
Udział w zajęciach dydaktycznych/praktyka	60 godz
Przygotowanie do zajęć	37 godz
Samodzielne studiowanie tematyki zajęć	51 godz
Egzamin lub kolokwium zaliczeniowe	2 godz
Sumaryczne obciążenie pracą studenta	150 godz
Punkty ECTS za moduł	6 ECTS

Pozostałe informacje

Szczegółowe treści kształcenia w ramach poszczególnych form zajęć (szczegółowy program wykładów i pozostałych zajęć)

Wykład

1. Programowanie dyskretne, programowanie całkowitoliczbowe, programowanie binarne, optymalizacja kombinatoryczna. Zastosowania – układanie rozkładów jazdy pociągów, planów lotów samolotów, sporządzanie harmonogramów prac dużych przedsięwzięć inwestycyjnych, planowanie produkcji, podejmowanie optymalnych decyzji.
2. Modele matematyczne wybranych problemów. Przegląd metod stosowanych do rozwiązywania zadań PC, klasyfikacja: modeli matematycznych, zagadnień praktycznych, metod numerycznych.
3. Ograniczenia w zadaniach PC. Liniowe układy diofantyczne, macierze unimodularne, postać Smitha i Hermita macierzy. Kraty ze skończonym zbiorem generatorów.
4. Twierdzenia o istnieniu rozwiązań liniowych układów diofantycznych. Macierze pseudoodwrotne. Metody rozwiązywania układów diofantycznych.
5. Wielościany wymierne, ściany wielościanów, wielościany całkowite. Macierze totalnie unimodularne, przedziałowe, zbalansowane. Twierdzenie Veinotta – Dantziga o związkach tych macierzy z wielościanami całkowitymi.
6. Twierdzenia Hoffmana—Kruskala o związkach macierzy wielościanami i grafami. Zadanie transportowe i przydziału.
7. Równanie dualne programowania liniowego. Totalna dualna całkowitość (TDC) (Edmonds, Giles). Bazy Hilberta. Twierdzenia o układach TDC i wielościanach całkowitych.
8. Wielomianowy algorytm testowania TDC. Grafy doskonałe i układy TDC. Twierdzenie Lovasza (bd).
9. Metoda podziału i ograniczeń Dakina dla PC i programowania zero—jedynkowego, przegląd pośredni, algorytm Balasa.
10. Zadanie plecakowe i załadunku. Problemy pokrycia, rozbicia i upakowania zbiorów.
11. Twierdzenia Gomory’ego—Chvatala o płaszczyznach odcinających, rząd Chvatala. Algorytmy płaszczyzn odcinających.
12. Zastosowania metody płaszczyzn odcinających, problem komiwojażera.
13. Metody geometrii algebraicznej. Pierścień wielomianów. Ideał. Twierdzenie Hilberta o zerach.
14. Bazy Groebnera. Algorytm Buchbergera. Zastosowania i kierunki rozwoju optymalizacji dyskretnej, problemy i nowe możliwości.

Ćwiczenia laboratoryjne

Program laboratoriów pokrywa się z programem wykładów

Rozwiązywanie problemów (głównie związanych z zagadnieniami praktycznymi) ilustrujących treści przekazywanych na kolejnych wykładach.

Metody i techniki kształcenia:

Wykład: Treści prezentowane na wykładzie są przekazywane w formie prezentacji multimedialnej w połączeniu z klasycznym wykładem tablicowym wzbogaconymi o pokazy odnoszące się do prezentowanych zagadnień.

Ćwiczenia laboratoryjne: W trakcie zajęć laboratoryjnych studenci samodzielnie rozwiązują zadany problem praktyczny, dobierając odpowiednie narzędzia. Prowadzący stymuluje grupę do refleksji nad problemem, tak by otrzymane wyniki miały wysoką wartość merytoryczną.

Warunki i sposób zaliczenia poszczególnych form zajęć, w tym zasady zaliczeń poprawkowych, a także warunki dopuszczenia do egzaminu:

Warunkiem koniecznym dopuszczenia do egzaminu jest posiadanie oceny pozytywnej z ćwiczeń. Dwa terminy zaliczeń poprawkowych są skorelowane czasowo z egzaminami poprawkowymi

Zasady udziału w poszczególnych zajęciach, ze wskazaniem, czy obecność studenta na zajęciach jest obowiązkowa:

Wykład:

- Obecność obowiązkowa: Nie

- Zasady udziału w zajęciach: Studenci uczestniczą w zajęciach poznając kolejne treści nauczania zgodnie z sylabusem przedmiotu. Studenci winni na bieżąco zadawać pytania i wyjaśniać wątpliwości. Rejestracja audiowizualna wykładu wymaga zgody prowadzącego.

Ćwiczenia laboratoryjne:

- Obecność obowiązkowa: Tak

- Zasady udziału w zajęciach: Studenci wykonują ćwiczenia laboratoryjne zgodnie z materiałami udostępnionymi przez prowadzącego. Student jest zobowiązany do przygotowania się w przedmiocie wykonywanego ćwiczenia, co może zostać zweryfikowane kolokwium w formie ustnej lub pisemnej. Zaliczenie zajęć odbywa się na podstawie zaprezentowania rozwiązania postawionego problemu. Zaliczenie modułu jest możliwe po zaliczeniu wszystkich zajęć laboratoryjnych.

Sposób obliczania oceny końcowej

1. Warunkiem koniecznym dopuszczenia do egzaminu jest posiadanie oceny pozytywnej z ćwiczeń.

2. Ocenę końcową **OK** wyznacza się na podstawie średniej ważonej **SW** obliczonej według wzoru

$$SW = 1/2 OL + 1/2 OE,$$

gdzie **OL** jest oceną uzyskaną z ćwiczeń laboratoryjnych,

a **OE** jest oceną uzyskaną z egzaminu.

3. Ocena końcowa **OK** jest obliczana według algorytmu:

Jeżeli $SW \geq 4.75$, to **OK** = 5.0 (bdb),

jeżeli $4.75 > SW \geq 4.25$, to **OK** = 4.5 (db),

jeżeli $4.25 > SW \geq 3.75$, to **OK** = 4.0 (db),

jeżeli $3.75 > SW \geq 3.25$, to **OK** = 3.5 (dst),

jeżeli $3.25 > SW \geq 3.00$, to **OK** = 3.0 (dst).

Sposób i tryb wyrównywania zaległości powstałych wskutek nieobecności studenta na zajęciach:

Student powinien zgłosić się do prowadzącego w celu ustalenia sposobu nadrobienia zaległości.

Wymagania wstępne i dodatkowe, z uwzględnieniem sekwencyjności modułów

Znajomość podstawowych faktów programowania liniowego (algorytm sympleks, dualność)

Zalecana literatura i pomoce naukowe

1. R. S. Garfinkel, G. L. Nemhauser, *Programowanie całkowitoliczbowe*, PWN Warszawa, 1978.
2. A. Schrijver, *Theory of Linear and Integer Programming*, Wiley, 1987.
3. M. M. Sysło, N. Deo, J. S. Kowalik, *Algorytmy optymalizacji dyskretnej*, PWN Warszawa, 1999.
4. S. Walukiewicz, *Programowanie dyskretne*, PWN Warszawa, 1986.
5. L.A. Wolsey, *Integer Programming*, Wiley, 1998.

Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu

1. Żak, Andrzej; General lower bound on the size of $(H;k)$ -stable graphs; *J. Comb. Optim.* 29, No. 2, 367-372 (2015).
2. Żak, Andrzej; On packing two graphs with bounded sum of sizes and maximum degree; *SIAM J. Discrete Math.* 28, No. 4, 1686-1698 (2014).
3. Żak, Andrzej; A generalization of an independent set with application to $(K_q;k)$ -stable graphs; *Discrete Appl. Math.* 162, Part 1, 421-427 (2014).
4. Żak, Andrzej; On embedding graphs with bounded sum of size and maximum degree; *Discrete Math.* 329, 12-18 (2014).
5. Żak, Andrzej; On $(K_q;k)$ -stable graphs; *J. Graph Theory* 74, No. 2, 216-221 (2013).
6. Żak, Andrzej; Near packings of graphs; *Electron. J. Comb.* 20, No. 2, Research Paper P36, 8 p., electronic only (2013).
7. Ruciński, Andrzej; Żak, Andrzej; Hamilton saturated hypergraphs of essentially minimum size; *Electron. J. Comb.* 20, No. 2, Research Paper P25, 16 p., electronic only (2013).
8. Żak, Andrzej; Growth order for the size of smallest Hamiltonian chain saturated uniform hypergraphs; *Eur. J. Comb.* 34, No. 4, 724-735 (2013).
9. Żak, Andrzej; Dudek, Aneta; On Hamilton-chain saturated uniform hypergraphs; *Discrete Math. Theor. Comput. Sci.* 14, No. 1, 21-28, electronic only (2012).
10. Görlich, Agnieszka; Żak, Andrzej; Sparse graphs of girth at least five are packable; *Discrete Math.* 312, No. 24, 3606-3613 (2012).
11. Cichacz, Sylwia; Görlich, Agnieszka; Nikodem, Mateusz; Żak, Andrzej; A lower bound on the size of $(H;1)$ -vertex stable graphs; *Discrete Math.* 312, No. 20, 3026-3029 (2012).

Informacje dodatkowe

Brak