

**AGH**AGH UNIVERSITY OF SCIENCE
AND TECHNOLOGY

Nazwa modułu zajęć: Algorytmy i struktury danych

Rok akademicki: 2019/2020 Kod: HNKT-1-210-s Punkty ECTS: 3

Wydział: Humanistyczny

Kierunek: Nowoczesne technologie w kryminalistyce Specjalność: —

Poziom studiów: Studia I stopnia Forma studiów: Stacjonarne

Język wykładowy: Polski Profil: Ogólnoakademicki (A) Semestr: 2

Strona www: —

Prowadzący moduł: dr inż. Frankowski Marek (mfrankow@agh.edu.pl)

Treści programowe zapewniające uzyskanie efektów uczenia się dla modułu zajęć

Celem przedmiotu jest zapoznanie studentów z najważniejszymi rodzinami algorytmów oraz powszechnie stosowanymi strukturami danych. Przedmiot wprowadza podstawy programowania obiektowego.

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	Student zna i rozumie algorytmy służące do rozwiązywania klasycznych problemów informatyki.	NKT1A_W05	Projekt, Kolokwium
M_W002	Student zna i rozumie podstawowe założenia programowania obiektowego i rozdziału silnik aplikacji - interfejs.	NKT1A_W05	Projekt, Kolokwium
Umiejętności: potrafi			
M_U001	Student potrafi projektować i implementować algorytmy operujące na złożonych strukturach danych.	NKT1A_U04, NKT1A_U06, NKT1A_U05, NKT1A_U07	Projekt, Kolokwium

M_U002	Student potrafi dobierać metody i narzędzia dostępne online do postawionych problemów programowania obiektowego i obsługi złożonych danych.	NKT1A_U06, NKT1A_U07	Projekt
Kompetencje społeczne: jest gotów do			
M_K001	Student potrafi prezentować zagadnienia podstaw programowania obiektowego, komunikować zasady działania złożonych algorytmów oraz odnosić się do dokumentacji problemu i opisów złożonych typów danych.	NKT1A_K04, NKT1A_K01, NKT1A_K02	Projekt, Aktywność na zajęciach

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
46	18	0	14	14	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	Student zna i rozumie algorytmy służące do rozwiązywania klasycznych problemów informatyki.	+	-	+	+	-	-	-	-	-	-	-
M_W002	Student zna i rozumie podstawowe założenia programowania obiektowego i rozdziału silnik aplikacji - interfejs.	+	-	+	+	-	-	-	-	-	-	-
Umiejętności: potrafi												
M_U001	Student potrafi projektować i implementować algorytmy operujące na złożonych strukturach danych.	+	-	+	+	-	-	-	-	-	-	-

M_U002	Student potrafi dobierać metody i narzędzia dostępne online do postawionych problemów programowania obiektowego i obsługi złożonych danych.	+	-	+	+	-	-	-	-	-	-	-
Kompetencje społeczne: jest gotów do												
M_K001	Student potrafi prezentować zagadnienia podstaw programowania obiektowego, komunikować zasady działania złożonych algorytmów oraz odnosić się do dokumentacji problemu i opisów złożonych typów danych.	+	-	+	+	-	-	-	-	-	-	-

Nakład pracy studenta (bilans punktów ECTS)

Forma aktywności studenta	Obciążenie studenta
Udział w zajęciach dydaktycznych/praktyka	46 godz
Przygotowanie do zajęć	14 godz
przygotowanie projektu, prezentacji, pracy pisemnej, sprawozdania	14 godz
Samodzielne studiowanie tematyki zajęć	10 godz
Dodatkowe godziny kontaktowe	5 godz
Sumaryczne obciążenie pracą studenta	89 godz
Punkty ECTS za moduł	3 ECTS

Pozostałe informacje

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

Wykład

1. Podstawy C1. I (podstawy)
2. Podstawy C2. II (funkcje)
3. Podstawy C3. III (klasy)
4. Podstawy C4. IV (klasy)
5. Podstawy programowania obiektowego (rozdział interfejs – silnik)
6. Kontenery (tablice i listy)
7. Dwuwymiarowe tablice i mapa (słownik)
8. Pliki tekstowe i wyjątki
9. Drzewa
10. Grafy I
11. Grafy II
12. Sortowanie I (różne algorytmy, porównanie czasu działania w różnych przypadkach danych, implementacja na tablicach i listach)

13. Sortowanie II (jak wyżej)

Ćwiczenia laboratoryjne

Ćwiczenia laboratoryjne zgodne z tematyką wykładów w formie instrukcji wprowadzających z możliwością konsultacji z prowadzącym.

Ćwiczenia projektowe

Zadania projektowe zgodne z tematyką wykładów, będące kontynuacją instrukcji wprowadzających realizowanych na zajęciach laboratoryjnych.

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.

Ćwiczenia projektowe: Nie określono

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

Aby uzyskać zaliczenie należy:

zdołać równo lub więcej niż 50% punktów z kartkówek

oraz

zdołać równo lub więcej niż 50% punktów z zadań domowych

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.

Ćwiczenia projektowe:

- Obecność obowiązkowa: Tak

- Zasady udziału w zajęciach: Nie określono

Sposób obliczania oceny końcowej

Ocena wyznaczana na podstawie regulaminu studiów z wagami - kartkówki 30% zadania domowe 70%. Aktywność na laboratoriach i wykładach może dodać +5% z wybranej części

Poprawy ocen:

Na ostatnich zajęciach można poprawić jedną najgorzej napisaną kartkówkę oraz oddać jedno najniższe ocenione zadanie (ocena zostaje podmieniona na nową) UWAGA: brak innych form zaliczeń poprawkowych

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

Nieobecności:

Nieusprawiedliwione → 0 z kartkówki

Usprawiedliwione → możliwość nadrobienia kartkówki w terminie określonym przez prowadzącego zajęcia

Zadanie domowe należy oddać na platformie terminowo bez względu na nieobecność, w tym usprawiedliwioną - przypadki skrajne rozstrzyga prowadzący laboratorium

Wymagania wstępne i dodatkowe, z uwzględnieniem sekwencyjności modułów

Umiejętność programowania w dowolnym języku imperatywnym.

Zalecana literatura i pomoce naukowe

Darmowe pomoce dostępne online:

<https://books.goalkicker.com/AlgorithmsBook/>

<https://books.goalkicker.com/CSharpBook/>

<https://docs.microsoft.com/pl-pl/dotnet/csharp/>

Publikacje naukowe osób prowadzących zajęcia związane z tematyką modułu

Nie podano dodatkowych publikacji

Informacje dodatkowe

Brak