

KARTA PRZEDMIOTU DLA NABORU 2024/2025 FORMA STUDIÓW: STACJONARNA/NIESTACJONARNA					
INFORMACJE OGÓLNE					
1. Przedmiot Podstawy programowania					
2. Wydział Nauk Technicznych					
3. Kierunek studiów Informatyka					
4. Poziom studiów studia pierwszego stopnia					
5. Liczba punktów ECTS 4					
6. Liczba godzin w semestrze					
semestr	w	ćw	lab/lek	prj/zp	prk
I	30/18		30/18		
7. Język wykładowy polski					
8. Wykładowca mgr inż. Maciej Hawryluk, mgr inż. Jarosław Wetoszka					
INFORMACJE SZCZEGÓŁOWE					
9. Wymagania wstępne					
1. Podstawowa umiejętność obsługi komputera					
10. Cele przedmiotu					
C1 Zapoznanie studentów z podstawowymi pojęciami z dziedziny programowania					
C2 Nauczenie studentów tworzenia algorytmów rozwiązujących proste problemy					
C3 Nauczenie studentów pisania programów					
C4 Zapoznanie studentów ze składnią języka C					
11. 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: algorytm, program.			K_W07, K_W11	
EU02	Zna i rozumie pojęcia: język programowania, kod źródłowy, kompilator.			K_W07, K_W11	
EU03	Zna i rozumie pojęcia: typ danych, stała, zmienna, identyfikator, operator.			K_W07, K_W11	
EU04	Zna i rozumie zasadę, dlaczego estetyka kodu i dobór identyfikatorów ma znaczenie, mimo że nie wpływa na działanie programu.			K_W11	
UMIEJĘTNOŚCI					
EU05	Potrafi tworzyć programy konsolowe wykonujące proste obliczenia matematyczne na podstawie danych wprowadzanych przez			K_U02, K_U15	

	użytkownika.	
EU06	Potrafi dobierać odpowiedni typ danych do deklaracji zmiennych wykorzystywanych w swoim programie.	K_U02, K_U15
EU07	Potrafi tworzyć programy, których przebieg zależy od decyzji podejmowanych przez użytkownika lub innych okoliczności sprawdzanych w trakcie działania programu.	K_U02, K_U15
EU08	Potrafi stosować instrukcje pętli w pisanych programach w celu wielokrotnego wykonania tych samych czynności.	K_U02, K_U15
EU09	Potrafi wykorzystywać tablice w celu przechowywania wielu danych tego samego typu i operowania na tych danych w programach.	K_U02, K_U15
EU10	Potrafi stosować podprogramy w pisany programie w celu dekompozycji złożonego algorytmu.	K_U02, K_U15
EU11	Potrafi wykorzystywać pliki do odczytu i zapisu danych w programach.	K_U02, K_U15
KOMPETENCJE SPOŁECZNE		
EU12	Jest gotów do odpowiedzialnego podejścia do pracy programisty poprzez rozumienie konsekwencji, które mogą wynikać z błędów znajdujących się w programach	K_K04
12. Treści programowe		
Forma zajęć – wykłady/laboratoria		
<p>Wykłady</p> <ol style="list-style-type: none"> 1) Podstawowe pojęcia. Pierwszy program. Instrukcja warunkowa 2) Deklaracje zmiennych. Liczbowe typy danych. Pętle. 3) Pobieranie danych z klawiatury. Napisy i znaki. Instrukcja wyboru. 4) Tablice jedno- i wielowymiarowe. 5) Podprogramy. Zmienne lokalne i globalne. 6) Podprogramy. Parametry podprogramów. 7) Odczyt i zapis danych do pliku. 8) Definiowanie własnych typów. Typy złożone. 9) Zmienne statyczne i dynamiczne. 10) Błędy w programach. <p>Laboratoria</p> <ol style="list-style-type: none"> 1) Zapoznanie z edytorem i kompilatorem. Pierwszy program – obliczanie obwodu i pola prostokąta. 2) Instrukcja warunkowa – program do obliczania objętości i pola całkowitego wybranych brył przestrzennych. 3) Pętle – program rysujący w konsoli wybrane figury geometryczne. 4) Instrukcja wyboru – konsolowy kalkulator z czterema działaniami i tekstowym menu. 5) Napisy – program, który zapamiętuje, wyświetla i łączy napisy. 6) Tablice jednowymiarowe – program do wykonywania obliczeń na ciągu liczb. 7) Tablice dwuwymiarowe – program do wykonywania obliczeń na macierzach. 8) Podprogramy – program wyświetlający ozdobne komunikaty na ekranie. 9) Podprogramy – program wykonujący proste obliczenia (np. potęgowanie). 10) Pliki – program odczytujący i zapisujący tekst do pliku. 11) Pliki – program odczytujący i zapisujący liczby do pliku. 12) Typy złożone – książka telefoniczna. 13) Wskaźniki – gra „Zgadnij liczbę”. 		
13. Narzędzia/metody dydaktyczne		
1. Wykłady w formie prezentacji		
2. Samodzielne tworzenie aplikacji przy użyciu Code::Blocks lub CLion		
3. Platforma Microsoft Teams		
4. Konsultacje		
14. Sposoby oceny (częstkowe, końcowe)		
1. Każde zajęcia laboratoryjne rozpoczynają się „wejściówką”.		

2. Każdy pisany przez studenta na laboratorium program jest oceniany.	
3. W połowie semestru przeprowadzane jest kolokwium z wykładu.	
4. W ciągu semestru studenci piszą samodzielnie w domu program, oceniany na koniec semestru.	
5. Na koniec semestru przeprowadzane jest kolokwium z wykładu.	
15. Obciążenie pracą studenta	
Forma aktywności	liczba godzin
1. Zajęcia z bezpośrednim udziałem nauczyciela oraz konsultacje	70/46
2. Nakład pracy studenta	30/54
suma	100
liczba punktów ECTS	4
16. Literatura	
Literatura podstawowa:	
1. Stephen Prata, <i>Język C. Szkoła programowania</i> . Wydanie VI, Helion, 2016.	
2. Marek Tłuczek, <i>Programowanie w języku C. Ćwiczenia praktyczne</i> . Wydanie II, Helion, 2011.	
3. Społeczność Wikibooks, <i>Programowanie w C</i> , Wikibooks, 2010. https://pl.wikibooks.org/wiki/C	
4. Greg Perry, Dean Miller, <i>Język C. Programowanie dla początkujących</i> . Wydanie III, Helion, 2016.	
Literatura uzupełniająca:	
1. L. Banachowski, K. Diks, W. Rytter, <i>Algorytmy i struktury danych</i> , Wydawnictwo Naukowe PWN, 2022.	
2. Maciej M. Sysło, <i>Algorytmy</i> , Helion, 2016.	
3. Peter Prinz, Tony Crawford, <i>C in a Nutshell: The Definitive Reference 2nd Edition</i> . O'Reilly Media, 2015.	
4. Brian W. Kernighan, Dennis M. Ritchie, <i>Język ANSI C. Programowanie</i> . Wydanie II. Helion, 2010.	
5. N. Wirth, <i>Algorytmy + Struktury Danych = Programy</i> , WNT, 2004.	
17. Formy oceny – szczegóły	
Warunki uzyskania zaliczenia przedmiotu: zajęcia kończą się zaliczeniem z oceną	
<u>Zaliczenie laboratorium</u>	
Każde zajęcia laboratoryjne rozpoczynają się 5-minutową wejściówką, zawierającą dwa pytania, sprawdzające przygotowanie studenta do zajęć. Wejściówka może zostać oceniona na 0, ½ lub 1 punkt: 0 – student przyszedł na zajęcia nieprzygotowany – ćwiczenie niezaliczone ½ – wejściówka zaliczona 1 – wejściówka zaliczona, ocena za ćwiczenie będzie podwyższona o pół stopnia Po wejściówce, każdy student pisze program, zgodnie ze specyfikacją przedstawioną przez prowadzącego. Program jest oceniany w skali od 2 do 5 (niektóre programy od 2 do 6). Studenci, którzy nie zdążyli skończyć programu na zajęciach, mogą go dokończyć w domu i oddać na kolejnych zajęciach (za każdy tydzień opóźnienia, ocena jest obniżana o pół stopnia). Dodatkowo, w drugiej połowie semestru studenci piszą w domu program, realizując zadany projekt – jest on oceniany na koniec semestru w skali od 2 do 6. Ocena końcowa z laboratorium jest wystawiana na podstawie średniej z ocen cząstkowych z zastrzeżeniem, że niezaliczenie więcej niż 3 ćwiczeń skutkuje oceną ndst oraz uwzględniając, że waga oceny za projekt jest 3-krotnie wyższa od wagi oceny za zwykły program.	
Średnia wymagana na poszczególne oceny:	
< 2,50 – 2.0 (ndst)	
2,50 - 3,24 – 3.0 (dst)	
3,25 - 3,74 – 3.5 (dst+)	
3,75 - 4,24 – 4.0 (db)	
4,25 - 4,74 – 4.5 (db+)	
> 4,74 – 5.0 (bdb)	
<u>Zaliczenie wykładu</u>	
W trakcie semestru studenci piszą dwa kolokwia, które sprawdzają ich wiedzę i umiejętności. Czas trwania każdego kolokwium to 30 minut. Większość pytań ma charakter otwarty. Punkty uzyskane z obu kolokwiów	

są sumowane i na tej podstawie jest wystawiana ocena na koniec semestru.

% uzyskanych punktów wymagany na poszczególne oceny:

0% - 50% – 2.0 (ndst)

50% - 59% – 3.0 (dst)

60% - 69% – 3.5 (dst+)

70% - 79% – 4.0 (db)

80% - 89% – 4.5 (db+)

90% - 100% – 5.0 (bdb)

18. 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 Białskiej im. Jana Pawła II
3. Zajęcia odbywać się będą zgodnie z aktualnym planem zajęć
4. Konsultacje odbywać się będą zgodnie z obowiązującym terminarzem