

# PAŃSTWOWA WYŻSZA SZKOŁA ZAWODOWA W NOWYM SĄCZU

## KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2013/2014

Instytut Techniczny

Kierunek studiów: Informatyka

Profil: Ogólnoakademicki

Forma studiów: Niestacjonarne

Kod kierunku: 11.3

Stopień studiów: I

Specjalności: Informatyka stosowana

### 1 PRZEDMIOT

NAZWA PRZEDMIOTU	Inżynieria oprogramowania A
KOD PRZEDMIOTU	IT 11.3 AIN B14 13/14
KATEGORIA PRZEDMIOTU	Przedmioty podstawowe i kierunkowe
LICZBA PUNKTÓW ECTS	8
SEMESTRY	4

### 2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

SEMESTR	WYKŁAD	ĆWICZENIA	LABORATORIUM	PROJEKT	SEMINARIUM
4	15		15	15	

### 3 CELE PRZEDMIOTU

**Cel 1** Umiejętność definiowania i wyjaśniania zasad skutecznego wytwarzania oprogramowania.

**Cel 2** Rozróżnia i komponuje procesy oraz metodyki produkcji i tworzenia oprogramowania oraz interpretuje i powiązuje języki inżynierii oprogramowania.

**Cel 3** Zilustruje i uzasadnia metody formalne, biblioteki i wzorce projektowe oraz zasady zarządzania projektami i konfiguracją oprogramowania oraz dostosuje i argumentuje techniki i narzędzia testowania oprogramowania.



## 4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

a Wiadomości podstawowe z matematyki wyższej.

## 5 EFEKTY KSZTAŁCENIA

**EK1** Wiedza: Ma uporządkowaną i podbudowaną teoretycznie wiedzę dotyczącą inżynierii oprogramowania, modelowania UML oraz zarządzania projektem. Ma wiedzę dotyczącą cyklu życia oprogramowania, a także urządzeń i systemów informatycznych.

**EK2** Umiejętności: Potrafi posługiwać się programami komputerowymi a także tworzy własne programy umożliwiające realizację określonych zadań.

**EK3** Kompetencje społeczne: Współpracuje w zespole.

## 6 TREŚCI PROGRAMOWE

### WYKŁAD

LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W1	Wstęp do inżynierii oprogramowania	1
W2	Zasady skutecznego wytwarzania oprogramowania	2
W3	Specyfikacja wymagań	1
W4	Kontrola jakości artefaktów	1
W5	Języki inżynierii oprogramowania	2
W6	Metody formalne	1
W7	Wzorce projektowe	1
W8	Zarządzanie konfiguracją	1
W9	Testowanie oprogramowania	2
W10	Projektowanie interfejsu użytkownika (API)	1
W11	Programowanie ekstremalne	1
W12	Ewolucja oprogramowania i refaktoryzacja	1
	RAZEM	15

### LABORATORIUM

LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
L1	Wprowadzenie do przedmiotu: Dowodzenie programu obliczającego n-ty wyraz ciągu Fibonacciego; Schematy programów, których poprawność udowodnić można wykorzystując podobnie jak w przypadku programów obliczających $n!$ i n-ty wyraz ciągu Fibonacciego; Opracowanie programu sprawdzającego czy dana liczba jest pierwsza w oparciu o schemat.	2
L2	Zasady skutecznego działania: Planowanie projektu a zasady skutecznego działania; Formulowanie celów i zadań metodą SMART; Praktyki zarządzania czasem; Technika wizualizacji i afirmacji.	2
L3	Specyfikacja wymagań.	2
L4	Metody formalne, sieci Petriego.	2
L5	Projektowanie i wzorce projektowe.	1
L6	Zarządzanie konfiguracją.	2
L7	Wprowadzenie do testowania - Pokrycie kodu. Automatyzacja wykonania testów.	2
L8	Programowanie Ekstremalne.	1
L9	Język UML.	1



## LABORATORIUM

LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
	RAZEM	<b>15</b>

## PROJEKT

LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
P1	Metodyki i procesy produkcji oprogramowania	1
P2	Metodyki tworzenia oprogramowania	1
P3	Narzędzia służące do zarządzania wymaganiami	1
P4	Narzędzia wspomagające zarządzanie zmianami (wymagań, kodu) w procesie produkcji oprogramowania	1
P5	Metryki dotyczące jakości oprogramowania	1
P6	Narzędzia generujące diagramy w UML-u	1
P7	Narzędzia CASE	1
P8	Technika zarządzania ryzykiem w projektach informatycznych	1
P9	Metody i techniki testowania oprogramowania	1
P10	Narzędzia do automatycznego testowania graficznego interfejsu użytkownika	1
P11	Metody estymacji wielkości kodu	1
P12	Narzędzia przeprowadzające statyczną analizę kodu źródłowego	1
P13	Style tworzenia kodu w wybranych językach programowania. Implementacja narzędzia do syntaktycznej kontroli stylu	1
P14	Narzędzia do tworzenia interfejsu użytkownika	1
P15	Podobieństwa i różnice metodyk Extreme Programming i Rational Unified Process	1
	RAZEM	<b>15</b>

## 7 METODY DYDAKTYCZNE

M1 Wykłady

M2 Prezentacje multimedialne

M3 Studium przypadku

M4 Ćwiczenia laboratoryjne

M5 Ćwiczenia projektowe



## 8 OBCIĄŻENIE PRACĄ STUDENTA

FORMA AKTYWNOŚCI	ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI
<b>Godziny kontaktowe z nauczycielem akademickim, w tym:</b>	
Godziny wynikające z planu studiów	45
Konsultacje przedmiotowe	13
Egzaminy i zaliczenia w sesji	2
<b>Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym:</b>	
Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	45
Opracowanie wyników	47
Przygotowanie raportu, projektu, prezentacji, dyskusji	48
<b>SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA</b>	<b>200</b>
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	8

## 9 SPOSOBY OCENY

### OCENA FORMUJĄCA

F1 Odpowiedź ustna

F2 Sprawozdanie z ćwiczenia laboratoryjnego

F3 Projekt indywidualny

F4 Projekt zespołowy

F5 Egzamin

### KRYTERIA OCENY

EFEKT KSZTAŁCENIA 1		MIEJSCE WERYFIKACJI	OPIS WERYFIKACJI EK 1
NA OCENĘ 3	Student ma wiedzę dotyczącą podstawowych metod inżynierii oprogramowania.	wykład, laboratorium, projekt	Ocena z odpowiedzi ustnych, ćwiczeń laboratoryjnych i projektowych. Egzamin.
NA OCENĘ 4	W stosunku do oceny 3,0 podwyższył swoją wiedzę o modelowanie i zarządzanie projektem.		
NA OCENĘ 5	W stosunku do oceny 4,0 podwyższył swoją wiedzę dotyczącą cyklu oprogramowania i urządzeń i systemów informacyjnych.		
EFEKT KSZTAŁCENIA 2		MIEJSCE WERYFIKACJI	OPIS WERYFIKACJI EK 2
NA OCENĘ 3	Przy pomocy nauczyciela akademickiego posługuje się programami komputerowymi dotyczącymi przedmiotu.	laboratorium, projekt	Ocena z ćwiczeń laboratoryjnych i projektowych.
NA OCENĘ 4	Samodzielnie posługuje się programami komputerowymi dotyczącymi przedmiotu.		



NA OCENĘ 5	Samodzielnie posługuje się programami komputerowymi dotyczącymi przedmiotu ponadto potrafi wykonywać własne programy umożliwiające realizację określonych zadań.		
EFEKT KSZTAŁCENIA 3		MIEJSCE WERYFIKACJI	OPIS WERYFIKACJI EK 3
NA OCENĘ 3	Potrafi wykonywać większość zadań przydzielonych przez zespół.	laboratorium, projekt	Ocena z ćwiczeń laboratoryjnych i projektowych.
NA OCENĘ 4	Posiada umiejętność wykonywania wszystkich zadań przydzielonych przez zespół.		
NA OCENĘ 5	Potrafi kierować pracą zespołu.		

**OCENA DO INDEKSU (OCENA PODSUMOWUJĄCA)**

Średnia ważona ocen cząstkowych uzyskanych za poszczególne efekty kształcenia.

**WARUNKI ZALICZENIA PRZEDMIOTU**

- a Zaliczenie zaplanowanych ćwiczeń laboratoryjnych i projektowych oraz pozytywnie zdany egzamin sprawdzający osiągnięcie założonych efektów kształcenia dla przedmiotu.

**10 MACIERZ REALIZACJI PRZEDMIOTU**

EFEKTY KSZTAŁCENIA DLA PRZEDMIOTU	ODNIESIENIE DO EFEKTÓW KIERUNKOWYCH	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	METODY DYDAKTYCZNE
EK1	INF_UO02, INF_W17, INF_UB02, INF_W14, INF_UP01, INF_UP02, INF_UP03, INF_UB04	Cel1, Cel2, Cel3	W1, W2, W3, W4, W5, W6, L1, L2, L3, L4	M1, M2, M3, M4
EK2	INF_UO02, INF_W17, INF_UB02, INF_W14, INF_UP01, INF_UP02, INF_UP03, INF_UB04	Cel1, Cel2, Cel3	W7, W8, W9, W10, W11, W12, L5, L6, L7, L8, L9	M1, M2, M3, M4
EK3	INF_UO02, INF_UB02, INF_UP01, INF_UP02, INF_UB04	Cel1, Cel2, Cel3	P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P11, P12, P13, P14, P15	M3, M5



## 11 WYKAZ LITERATURY

### LITERATURA PODSTAWOWA:

- [1] Beynon-Davies P. — *Inżynieria systemów informacyjnych*, Warszawa, 2004, WNT
- [2] Borowik B., Kurytnik I. — *Mikrokontrolery PIC w zastosowaniach*, Warszawa, 2009, PAK
- [3] Górski J. — *Inżynieria oprogramowania w projekcie programistycznym*, Warszawa, 2000, MIKOM
- [4] Kan S. H. — *Metryki i modele w inżynierii jakości oprogramowania*, Warszawa, 2006, PWN
- [5] Sommerville I. — *Inżynieria oprogramowania, Seria: Klasyka informatyki*, Warszawa, 2003, WNT

### LITERATURA UZUPEŁNIAJĄCA:

- [1] Beck K., Cyntia A. — *Wydatne programowanie - Extreme Programming*, Warszawa, 2005, MIKOM
- [2] Borowik B. — *Meandry Języka C*, Warszawa, 2005, PWN
- [3] Fowler M., Scott K. — *UML w kropelce, wersja 2.0*, Warszawa, 2005, LTP Oficyna
- [4] Warmer J., Kleppe A. — *Inżynieria oprogramowania OCL, precyzyjne modelowanie w UML*, Warszawa, 2003, WNT

## 12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

### OSOBA ODPOWIEDZIALNA ZA KARTĘ

prof. dr hab. inż. Mikołaj Karpiński (kontakt: mkarpinski@ath.bielsko.pl)

### OSOBY PROWADZĄCE PRZEDMIOT

prof. dr hab. inż. Mikołaj Karpiński (kontakt: mkarpinski@ath.bielsko.pl)

dr inż. Ihor Pazdriy (kontakt: irpazdriy@gmail.com)

mgr inż. Józef Wójcik (kontakt: j.wojcik@pwsz-ns.edu.pl)

## 13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejscowość, data)	(odpowiedzialny za przedmiot)	(kierownik zakładu)	(dyrektor instytutu)
---------------------	-------------------------------	---------------------	----------------------

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....  
.....  
.....