

# Programowanie funkcyjne

## Wymagania wstępne:

Podstawy programowania, podstawy logiki.

## Program przedmiotu:

1. Charakterystyka programowania funkcyjnego.
2. Podstawowe elementy programowania funkcyjnego na przykładzie wybranego języka.
3. Przegląd typów danych: typy wbudowane, wzorce i konstruktory, produkty kartezyjańskie, listy, rekordy.
4. Wyrażenia logiczne i wyrażenia warunkowe.
5. Przykłady procedur wyższego rzędu.
6. Zagadnienie i przykłady polimorfizmu.
7. Listy w programowaniu funkcyjnym.
8. Rekurencja i obsługa wyjątków.
9. Konstrukcje imperatywne: referencje, tablice.
10. Wprowadzenie do programowania w języku Haskell.
11. Wzorce w języku Haskell.
12. Klasy i moduły w języku Haskell

## Metodyka zajęć:

Wykład z wykorzystaniem środków audiowizualnych. Na ćwiczeniach wprowadzane zostaną stopniowo elementy omawiane na wykładzie. Sugerowane jest przedstawienie dwóch wybranych języków ( w tym jeden to Haskell).

## Cel dydaktyczny przedmiotu:

Zapoznanie studentów z funkcyjnym paradygmatem programowania.

## Forma zaliczenia: zaliczenie

## Literatura:

1. Ch. Reade, Elements of Functional Programming, Addison-Wesley, 1993.
2. Smith, Chris, Programming F#. O'Reilly, 2009
3. Graham Hutton: Programming in Haskell Cambridge University Press, 2007
4. H. Abelson, G. J. Sussman, Struktura i interpretacja programów komputerowych, WNT 2002
5. Simon Thompson: Haskell: The Craft of Functional Programming, Second Edition, Addison-Wesley, 1999.