

MI008	Semantika programskih jezika	P	V	S	ECTS 6
		2	2	0	

**Cilj predmeta.** Upoznati studente s apstraktnim modelom programskih jezika  $\lambda$ -calculus. Objasniti generalna svojstva programa koji može mijenjati druge programe u apstraktnom okruženju neopterećenog detaljima. Ovladati pojmovima varijabla, apstrakcija (funkcijska definicija), primjena funkcije na argument (Application). Povezati ove koncepte s konceptima funkcijskog programiranja i jezika Haskell.

**Potrebna predznanja.** Haskell ili neki drugi funkcijski jezik.

### Sadržaj predmeta.

1. Uvod. Lambda notacija. Aksiomska, operacijska i denotacijska semantika.
2. Jezik PCF (Programming Computable Functions). Funkcijsko programiranje i tipirani lambda račun (typed lambda calculus). Boolean, prirodni broj, uparivanje i funkcijski izrazi. Definicija rekurzivnih funkcija uz pomoć operatora fiksne točke. PCF redukcija i simbolički interpreteri. Rekord i  $n$ -torke. Iteracija i repna (tail) rekurzija. Potpuno rekurzivne funkcije i parcijalno rekurzivno funkcije.
3. Univerzalna algebra i algebarski tipovi podataka (equational proof system, soundness, completeness). Aspekti algebarske teorije tipova podataka. Homomorfizam i izomorfizam.
4. Semantika tipiranog lambda računa i rekurzije (context-sensitive syntax by typing rules, general models, domain-theoretic models).
5. Imperativni programi. Sintaksa While“ programa. L-vrijednosti i R-vrijednosti. Strukturirana operacijska semantika.

### ISHODI UČENJA

R.b.	ISHODI UČENJA
1.	Definirati lambda notaciju.
2.	Opisati sintaksu i semantiku tipiranog lambda računa.
3.	Opisati razlike između aksiomske, operacijske i denotacijske semantike.
4.	Interpretirati metode standardnog programiranja u kontekstu funkcijskog jezika.
5.	Objasniti algebarske izraze i njihovu interpretaciju u multi-sortiranim (multi-sorted) algebra.
6.	Objasniti ekvivalentnost aksiomske i denotacijske semantike.
7.	Razlikovati dvije vrste modela u kontekstu tipiranog lambda računa; parcijalno uređene strukture, rekurzivna funkcijska teorija.

### POVEZIVANJE ISHODA UČENJA, ORGANIZACIJE NASTAVNOG PROCESA I PROCJENA ISHODA UČENJA

ORGANIZACIJA NASTAVNOG PROCESA	ECTS	ISHOD UČENJA **	AKTIVNOST STUDENATA*	METODA PROCJENE	BODOVI	
					min	max
Pohađanje predavanja i vježbi	1	1-7	Prisutnost na nastavi, rasprava, timski rad, samostalan rad na zadacima i kratke provjere znanja	Potpisne liste, praćenje aktivnosti na nastavi, zadaci zatvorenog tipa	0	4
Domaće zadaće	1	1-7	Samostalno rješavanje programerskih zadataka	Provjera točnih rješenja (ocjenjivanje)	0	4
Provjera znanja (kolokvij)	2	1-7	Priprema za pismenu provjeru znanja	Provjera točnih odgovora (ocjenjivanje)	25	46
Završni ispit	2	1-7	Ponavljanje gradiva	Usmeni ispit	25	46
UKUPNO	6				50	100

**Izvođenje nastave i vrednovanje znanja.** Predavanja i vježbe su obvezni. Ispit se sastoji od pismenog i usmenog dijela, a polaže se nakon odslušanih predavanja. Prihvatljivi rezultati postignuti na kolokvijima, koje studenti pišu tijekom semestra, zamjenjuju pismeni dio ispita. Vježbe su auditorne. Studenti mogu utjecati na ocjenu tako da tijekom semestra pišu domaće zadaće.

**Može li se predmet izvoditi na engleskom jeziku:** Da

**Osnovna literatura:**

1. J. C. Mitchell, Foundations for Programming Languages, MIT Press, 1996.

**Dopunska literatura:**

1. J. R. Hindley, J. P. Seldin, Lambda-Calculus and Combinators, an Introduction, Cambridge University Press 2008.
2. F. Baader, T. Nipkow, Term Rewriting and All That, Cambridge University Press 1998.
3. M. R. A. Huth, M. D. Ryan, Logic in Computer Science, Modelling and Reasoning about Systems, Cambridge University Press, 2000.