

<b>Студијски програм:</b> ОАС Рачунарске науке, ОАС Информационе технологије			
<b>Назив предмета:</b> Програмски језици и парадигме			
<b>Наставник/наставници:</b> Милош Савић			
<b>Статус предмета:</b> изборни			
<b>Број ЕСПБ:</b> 7			
<b>Услов:</b> нема			
<b>Циљ предмета</b> Циљ предмета је да студенте упозна са различитим програмским парадигмама (изуевши објектно-орјентисану која је покривена другим курсевима на студијском програму), основним концептима и конструктима функционалног и мултипарадигматског стила програмирања и конкретним програмским језицима који су њихови карактеристични представници.			
<b>Исход предмета</b> <i>Минимални:</i> Минимално се очекује да студент разуме концепте и конструкте различитих програмских парадигми, те схвати значај различитих стилова програмирања. <i>Пожељни:</i> На крају курса студент је оспособљен да развија софтверска решења користећи функционалне и мултипарадигматске програмске језике и поседује одговарајуће аналитичке способности да на основу постављених захтева одабере најадекватнију парадигму за имплементацију неког софтверског система.			
<b>Садржај предмета</b> <i>Теоријска настава</i> Поделе програмских језика и кључне програмске парадигме. Основне разлике између процедуралне, функционалне и логичке парадигме. Дефиниција програмског језика: лексика, синтакса и семантика. Системи типова програмских језика: статички/динамички типизирани језици, слабо/јасно типизирани језици, ад-хок полиморфизам и преоптерећивање оператора, статичко/динамичко везивање, генерички типови, параметријски полиморфизам, хијерархијски полиморфизам, еквиваленција типова, механизми провере и извођења типова. Основне карактеристике функционалног стила програмирања: апстракција тока извршавања, референцијална транспарентност, функције вишег реда, стриктна и не-стриктна семантика, бесконачне структуре података, ламбда рачун и анонимне функције, парцијално примењене функције, окружења и затворења. Приказ могућности једног функционалног програмског језика (нпр. Хаскел). Рекурзивне функције, упаривање образаца и процесирање листи. Декларације типова и класе типова. Мултипарадигматски програмски језици и приказ могућности једног таквог језика (нпр. Раст или Скала). Фузија објектно-орјентисаног и функционалног стила програмирања. Референцијални типови и паметни показивачи, алокација и деалокација меморије, основни конструкти за конкурентно програмирање. <i>Практична настава</i> Решавање конкретних програмских проблема (имплементације алгоритама и структура података) у функционалним и мултипарадигматским програмским језицима користећи њихове карактеристичне конструкте.			
<b>Литература</b> 1. Maurizio Gabbriellini and Simone Martini. Programming Languages: Principles and Paradigms, Springer, 2010. 2. Graham Hutton. Programming in Haskell, Cambridge University Press, 2016. 3. Martin Odersky Lex Spoon and Bill Venners. Programming in Scala, Addison-Wesley, 2016. 4. Steve Klabnik and Carol Nichols. The Rust Programming Language, 2nd Edition, 2022.			
<b>Број часова активне наставе</b>	<b>Теоријска настава: 2</b>	<b>Практична настава: 3</b>	
<b>Методe извођења наставе</b> Предавања су базирана на класичном моделу извођења наставе који укључује презентовање наставног материјала коришћењем пројектора. На вежбама, које се одвијају у рачунарској лабораторији, студенти решавају конкретне програмске задатке у функционалним и мултипарадигматским програмским језицима. Да би изашао на усмени испит студент треба да положи предиспитне обавезе које се састоје од израде два практична колоквијума на рачунару (један колоквијум покрива функционално, а други мултипарадигматско програмирање). На усменом испиту се очекује да студент демонстрира разумевање теоријских концепата обухваћених курсом.			
<b>Оцена знања (максимални број поена 100)</b>			
<b>Предиспитне обавезе</b>	поена	<b>Завршни испит</b>	поена
Два практична колоквијума	50 (25 + 25)	Усмени испит	50