

Ispit iz programskih paradigmi

18.6.2015

1. **[Prolog 33%]** Napisati predikat koji ispituje da li su cifre datog broja tako složene da imaju tačno jedan lokalni maksimum nad spiskom svih cifara tog broja.

Prvi(45322):-true. //lokalni maksimum je samo u petici

Prvi(123321):-false. //lokalni maksimumi su u dve trojke

Prvi(123212):-false. //lokalni maksimumi su u trojci i u poslednjoj dvojci

2. **[Haskell 34%]** Napisati funkciju koja na osnovu zadate liste podlisti realnih brojeva i zadatog broja K formira listu koja sadrži samo one podliste koje imaju prosečnu vrednost veću od realnog broja K. Dodatno, kao poslednji element svake od tih podlisti se dodaje dobijena prosečna vrednost. Obavezna je upotreba bar jedne ugrađene funkcije višeg reda.

Drugi $[[5, 6, 7],[1.2, 5, 7],[1, 1.9, 2.8, 3],[5, 6, 8, 4, 3]]$ 5.1 = $[[5,6,7,6],[5,6,8,4,3,5.2]]$

3. **[Python 33%]** Napisati skript koji otvara lokalnu tekstualnu datoteku **razno.txt**, ekstrahuje iz nje sve identifikatore, i njihova **jedinstvena** pojavljivanja upisuje u drugu datoteku **identifikatori.txt**. Pod identifikatorom podrazumevati reči koje sadrže samo slova, cifre i podvlaku, pri čemu kao prvi karakter može da figuriše samo slovo ili podvlaka.

razno.txt: fdsal 343 ekfa34 34afad _dfsl fdsal identifikatori.txt: fdsal ekfa34 _dfsl

Vreme za rad: 120 minuta. Srećno!

Ispit iz programskih paradigmi

18.6.2015

1. **[Prolog 33%]** Napisati predikat koji ispituje da li su cifre datog broja tako složene da imaju tačno jedan lokalni maksimum nad spiskom svih cifara tog broja.

Prvi(45322):-true. //lokalni maksimum je samo u petici

Prvi(123321):-false. //lokalni maksimumi su u dve trojke

Prvi(123212):-false. //lokalni maksimumi su u trojci i u poslednjoj dvojci

2. **[Haskell 34%]** Napisati funkciju koja na osnovu zadate liste podlisti realnih brojeva i zadatog broja K formira listu koja sadrži samo one podliste koje imaju prosečnu vrednost veću od realnog broja K. Dodatno, kao poslednji element svake od tih podlisti se dodaje dobijena prosečna vrednost. Obavezna je upotreba bar jedne ugrađene funkcije višeg reda.

Drugi $[[5, 6, 7],[1.2, 5, 7],[1, 1.9, 2.8, 3],[5, 6, 8, 4, 3]]$ 5.1 = $[[5,6,7,6],[5,6,8,4,3,5.2]]$

3. **[Python 33%]** Napisati skript koji otvara lokalnu tekstualnu datoteku **razno.txt**, ekstrahuje iz nje sve identifikatore, i njihova **jedinstvena** pojavljivanja upisuje u drugu datoteku **identifikatori.txt**. Pod identifikatorom podrazumevati reči koje sadrže samo slova, cifre i podvlaku, pri čemu kao prvi karakter može da figuriše samo slovo ili podvlaka.

razno.txt: fdsal 343 ekfa34 34afad _dfsl fdsal identifikatori.txt: fdsal ekfa34 _dfsl

Vreme za rad: 120 minuta. Srećno!