

Виктор Василев 211504

**Програмски парадигми
Домашна задача 2**

Задача 1

Под коментарот shared utils имаме предикати кои се среќаваат во повеќе задачи и ги реискористуваме повеќе пати. Тоа се предикатот my_concat/3 за спојување на две листи, br_el/2 за наоѓање на број на елементи на листа, lista_sodrzy/2 која проверува дали листа содржи даден елемент, najdi_roditeli/3, кој наоѓа мајка и татко на дадено лице, kolku_pati_vo_lista/3 кажува колку пати даден елемент се појавува во дадена листа, zemi_prv/2 кој зема прв елемент од листа.

Задача 1а

Предикатот zemi_grad_raganje/2 го враќа градот за дадена шифра на лице. reshenie/1 проверува дали лицето со дадената шифра ги исполнува условите кои ги бара задачата, односно дека е роден во различен град од градовите на раѓање на неговите мајка и татко. rodeni_razlicen_grad/1 ни ја дава бројката колку вакви лица постојат кои ги исполнуваат овие услови.

Задача 1б

Предикатот den_vo_godina/3 наоѓа реден број за даден ден во годината. На пример - 25ти февруари е: $30 * 1 + 25 = 55$ (30 бидејќи земаме дека сите месеци имаат 30 дена). razlika_denov/3 ја наоѓа абсолютанта разлика во денови помеѓу два бројки што претставуваат денови. predci_basic/3 ни е потребен за барање на предците кои се исти род како лицето и нивната разлика во родендени да е помала или еднаква на 7 дена. predci е финалниот предикат кој го користиме за решавање на задачата.

Задача 2а

Дефинираме предикат zemi_ime_i_prezime_na_broj за да го најдеме името и презимето за даден телефонски број. broj_povici_od_mene/2 го наоѓа бројот на повици кои потекнуваат од даден број. site_broevi_na_koi_im_e_zvoneto/1 ги наоѓа сите броеви кои се во листите на повици, односно броеви на кои што некој им се јавил. vkupno_povici_na_broj/2 го наоѓа бројот на повици во кои учествувал даден број (тој е caller или callee). Предикатот koj_ima_max_vkupno_povici/3 го наоѓа бројот кој има најголем број на повици во кои учествувал и najbroj/2 го дава неговото име и презиме.

Задача 2б

Предикатот `povici_od_mene/2` ги наоѓа сите повици кои потекнуваат од даден број но трансформирани во форма `povik(Sender, Receiver, 100)`. `povici_kon_mene/2` ги наоѓа сите повици кои се упатено кон даден број но трансформирани во форма `povik(Sender, Receiver, 100)`. `smsi_prateni_od/2` ги наоѓа сите смс пораки пратени од даден број, но трансформирани во `povik(Sender, Receiver, 100)`. Сличното се случува со `smsi_prateni_kon/2` но за смс пораки кои се пратени ДО дадениот број. `site_povici_na/2` ги наоѓа сите повици и смс пораки во кои е вклучен даден број како `povik(sender, receiver, duration)`. `total_time_vo_lista_so_broj/3` го наоѓа вкупното време кое даден број (чија листа на сите повици ја даваме како параметар) го потрошил со `Broj2`. Предикатот `site_telefoni/1` ги зема сите телефонски броеви во една листа. `od_povik_zemi_toj_sto_ne_e_broj/3` работи така што од `povik(Sender, Receiver, Duration)` го зема `Sender` или `Receiver` во зависност од тоа кој НЕ е еднаков на `Broj` (параметар). `omilen_inner/4` го наоѓа бројот со кој дадениот број има најмногу потрошено време и `omilen/2` е решението на задачата и е само `wrapper` околу `omilen_inner/4`.

Задача 3а

Предикатот `izbroj_lokacija/2` е решението на задачата и наоѓа колку пати дадената локација била почетна или крајна за некоја услуга.

Задача 3б

`distance/3` наоѓа било кое растојание помеѓу две локации. `shortest_path/3` наоѓа најкраток пат помеѓу две локации, `path_length/4` е `dijkstra bfs loop` за кои чуваме посетени локации. `client_distance` наоѓа за даден клиент вкупно растојание кое го поминал возејќи се со такси. `najmnogu_kilometri/2` е решението на задачата и го наоѓа клиентот кој поминал најмногу вкупно растојание и ги враќа неговото име и презиме.

Задача 3в

Предикатот `presmetaj_cena_voznja` наоѓа колку чини возење од една локација до друга имајќи на ум дека таксито секогаш го одбира најкраткиот пат. `cena_za_taksi_broj`, ја зема цената на километар за даден такси број. `dali_e_iskoristen/1` проверува дали даден број припаѓа на такси возило со кое некој клиент се има возено. `total_cena/2` е вкупната цена која даден такси број ја заработил. `generate_total_cena/1` е предикат за генерирање на сите вкупни заработувачки на секое такси возило кое е искористено. `najmnogu_zarabotil/1` е решението на задачата и наоѓа кој такси број заработил најмногу.