Teme:

- uporaba že zgrajenih razredov,
- osnove kriptografije – frekvenčna analiza,
- ponovitve: datoteke, nizi, tabele;
- raba kolekcij za poenostavitev operacij,
- UML diagrami.

Vaje je osnovana na primere, izvedenem (prepisi: 20160302_f_analiza) med uro teoretičnih vsebin. Cilj vaje je spisati aplikacijo za dekripcijo kriptiranega besedila na osnovi analize pogostosti črk v besedilu. Podane zahteve in nekaj prototipov se nahaja v datoteki 11_RSO04_vaja_priponka.zip., v isti datoteki boste našli tudi urejen primer, ki je bil izveden pri uri teoretičnih vsebin.

Naloga 1

Realizirajte zahtevano aplikacijo, podano s predpisi in opisi. Rezultat dekripcije mora biti datoteka, podana s specifikacijo. Za enkripcijski mehanizem lahko uporabite metodo o2c ali pa postopek RailFence (imate ga v prepisih)

Naloga 2

Skušajte dekriptirati kriptirano besedilo, spisano v angleščini. Kaj opazite pri dekripciji?

Naloga 3

Sestavite uporabniški vmesnik v tehnologiji JavaFX, ki bo omogočal zajeti zahtevana besedila, izbiro jezika, viszualizacijo kriptiranih in dekriptiranih datotek.

Naloga 4

Dopolnite program tako, da bo omogočaj shranjevanje frekvenc črk za posamezne jezike in po potrebi tudi nalaganje frekvenc iz izbrane datoteke, hkrati pa dodatno 'učenje' v primeru, da vizualno opazite, da dekripcija ni dovolj dobra.

Naloga 2

Poslovenite protokol in ga prilagodite (naj spraševanje po čemerkoli). Pri tem ga raztegnite tako, da bo izvedel vsaj 8 korakov konverzacije.

Naloga 3

Popravite realizacijo strežnika tako, da se bodo tudi v oknu strežnika izpisovali odzivi odjemalca in zahteve strežnika, kot se to dogaja v odjemalcu.

Naloga 4

V nalogi smo pripravili osnovo za dvotočkovno povezavo (P2P) med strežnikov in odjemalcem. Strežnik ima trenutno s protokolom »programiran« odziv. Za dejanski klepet potrebujemo tudi pri strežniku poljuben odziv oz. interakcijo z »uporabnikom«. Odstranite programiran odziv in namesto njega izvedite branje odziva strežnika s tipkovnice.

Naloga 5

Preoblikujete aplikacijo odjemalca v vizualno aplikacijo. Ta naj omogoča vnos naslova strežnika in komunikacijskih vrat, pričetek in konec povezave. Konverzacija med strežnikom in odjemalcem naj se beleži v besedilnem oknu.

Ker je bila večina zadanih vaj predhodno opravljena kot demonstracija pri urah teoretičnih vsebin, so zgolj spodnje obvezne za oddajo:

Naloga 6

Realizirajte „logiranje" oz. dnevniško datoteko konverzacije med strežnikom in odjemalcem/odjemalci (predlagam besedilno datoteko, pri čemer enostavno dodajate vrstice na konec datoteke za vsak odziv 'strežnika' in vsak odziv 'odjemalca' ('log' za strežnik in 'log' za odjemalca). Dodajte tudi čase priklopa in odklopa, če je to možno.

Naloga 7

Popravite odjemalca tako, da bo omogočal zajem podatkov o naslovu strežnika in vratih strežnika. V primeru, da je povezava s strežnikom vzpostavljena, ne omogočite še ene povezave (gumb poveži). Dodajte še zajem identifikacije uporabnika, ki bo uporabljena pri konverzaciji.

Naloga 8

Dodajte strežniku protokol, ki bo omogočal: odklop odjemalca, zajem uporabnikove identitete in izpis te identitete pri konvezaciji (trenutno se uporablja črka 'mark'). Popravite odjemalca tako, da bo upošteval protokol strežnika.

Naloga 9*[i]

'Popravite' strežnik tako, da bo omogočal priklop poljubnega števila odjemalcev, ki bodo vsi komunicirali med seboj. V protokol dodajte še možnost pošiljanja zgolj izbranemu odjemalcu. Npr.: PRIVATE TO joco: kaj se pa greš? Pošlje besedilo zgolj odjemalcu z identiteto 'joco'. PRIVATE TO je identifikacija privatnega sporočila, če ne sledi aktiven 'joco' se tako sporočilo pač zavrže.

[i] Rahlo težja naloga (neobvezna v primeru, da ja vaš domet pod pravdobro)