

## Alberta koledža

Studiju kurss “Datu struktūras un algoritmi”

vieslektors Egmonts Treiguts

Patstāvīgo studiju 3. kontroldarbs

Izpētīt koku datu struktūras un to realizācijas metodes. Izstrādāt vienkāršotas failu sistēmas demonstrācijas lietojumprogrammu C++ OOP valodā .

- 1) Izveidot koka objekta klasi.
- 2) Izveidot koka mezgla klasi.
  - a) Katram koka mezglam ir tā veids: “katalogs” vai “fails”.
  - b) Jebkurā katalogā var būt jebkurš apakškatalogu vai failu skaits.
  - c) Par koka saknes mezglu uzskatīt loģiskā diska pamatkatalogu.
  - d) Jebkuram mezglam ar veidu “katalogs” var būt apakšmezgli: citi katalogi vai faili.
  - e) Mezglam “fails” apakšmezglu nevar būt. Mezglis “fails” ir koka lapa.
  - f) Katram mezglam paredzēt šādus datus: unikals mezgla identifikators, augstāka līmeņa mezgla identifikators, mezgla nosaukums, mezgla veids.
  - g) Mezgla nosaukumā nedrīkst būt komata (,) simbols.
- 3) Uzrakstīt galvenās metodes darbam ar koku:
  - a) Jauna mezgla pievienošana mezglam ar zināmu identifikatoru;
  - b) Koka zara attēlošanu uz ekrāna kā daudzlīmeņa sarakstu attēlojot mezgla identifikatoru, nosaukumu un tā veidu. Piemēram:  
Folder (4) – Darba\_faili  
    File (7) – image.jpg  
    Folder (8) – Ligumi  
        File (12) – Darba\_ligums.xls  
        File (18) – Ligums\_internets.docx
  - c) Mezgla dzēšana pēc tā identifikatora. Ja mezglis nav lapa, tad nodzēst visu zaru (rekursīvs algoritms).
- 4) Demonstrācijas piemērā izveidot gatavu koka struktūru ar vismaz trijiem līmeņiem.
  - a) Programmas lietotāja saskarne (komandas):
    - i) h[Enter] – lietojumprogrammas komandrindas lietošanas pamācība.
    - ii) +id,augstaka\_līmeņa\_mezgla\_id,nosaukums,veids[Enter] – pievienot mezglam ar noteiktu identifikatoru “augstaka\_līmeņa\_mezgla\_id” jaunu mezglu. Ja mezglis ar “id” identifikatoru jau eksistē un tā veids sakrīt ar pievienojamā mezgla veidu, tad esošo mezglu aizvietot ar jauno ignorējot “augstaka\_līmeņa\_mezgla\_id”. Ja veids nesakrīt, tad paziņot par kļūdu. Veida norādīšanai lietot apzīmējumus, piemēram “d” – katalogam, “f” – failam.
    - iii) -id[Enter] – nodzēst mezglu ar doto “id”. Ja šāds mezgla neeksistē, tad paziņot par kļūdu. Ja mezgla nav lapa, tad nodzēst visu zaru (rekursīvs algoritms).

- iv) \*id[Enter] – attēlot koka zaru sākot ar “id” mezglu.
  - v) \*[Enter] – attēlot visu koku.
  - vi) \$[Enter] – beigt programmu;
- b) Paredzēt kļūdu apstrādi ar atbilstīgiem paziņojumiem:
- i) Ja pirmais ievadītais simbols nav [h+\*]\$ – nepazīstama komanda.
  - ii) Ja kataloga vai faila nosaukuma garums ir 0.
  - iii) Ja elementa veids nav ne katalogs, ne fails.
  - iv) Ja mezglā ar norādīto id nav atrasts.

**Norādījumi:**

- Programmēšanas valoda C++.
- Kodam ir jābūt ar atbilstīgiem komentāriem.
- Projekta katalogu arhivēt (ZIP) un ielādēt e-vidē.