

## 4. pielikums

### Vērtēšanas uzdevumu piemēri

#### 1. uzdevums

#### Instrumentu un tehnikas uzskaites sistēma

#### (1. temats. Objektorientēta programmēšana un ārējās bibliotēkas)

**Sasniedzamais rezultāts:** veido programmu vienā no objektorientētajām programmēšanas valodām; izmanto programmēšanas valodas un tās bibliotēku dokumentāciju un palīdzības sistēmu, lai patstāvīgi apgūtu citas to piedāvātās iespējas, kas nepieciešamas konkrētās programmatūras izstrādei.

#### Uzdevums

Izlasi problēmas aprakstu un specifikāciju un, ievērojot objektorientētās programmēšanas valodas pamatprincipus, uzraksti programmu, kas risina doto problēmu!

**Problēmas apraksts:** Kārlim ir savs uzņēmums, kas piedāvā nomāt dažādus celtniecības instrumentus un tehniku. Līdz šim iznomāto produktu uzskaiti Kārlis veica pierakstu kladē.

Pieaugot klientu un nomas pieprasījumu skaitam, Kārlim kļuvis grūtāk veikt instrumentu un tehnikas uzskaitījumu kladē. Informācija ir kļuvusi nepārskatāma.

Kāds paziņoja Kārlim ieteica automatizēt nomas instrumentu un tehnikas uzskaiti datorā. Kārlis vērsās pēc palīdzības pie programmētājiem, un, vadoties pēc viņa uzskaites principiem pierakstos, tika sastādīta aptuvena specifikācija vēlamo programmatūras risinājumam.

**Specifikācija par nepieciešamajām klasēm, konstruktoriem, metodēm, lietotāja saskarni, datu glabāšanu:**

Programma sevī ietver klasi (*class*) **Noma**, kurā ir īpašības (*property/field*):

- **Produkta\_kategorija** (piemēram, urbjmašīna, zāģis)
- **Produkta\_nosaukums** (piemēram, Bosch GSR 18)

- **Tehniskie\_raksturojumi** (piemēram, akumulators: 2 gab. / 18 V / 5,0 Ah / Li-ion)
- **Nomas\_cena\_dienā** (piemēram, 15 EUR)
- **Produkts\_pieejams** (piemēram, jā/nē)
- **Nomnieks:\_vārds** (piemēram, Andrejs)
- **Nomnieks:\_uzvārds** (piemēram, Priede)
- **Nomnieks:\_p.k.** (piemēram, 123456-78901)
- **Nomnieks:\_tel.\_numurs** (piemēram, 26543218)
- **Nomas\_sākuma\_datums** (piemēram, 07.12.2020.)
- **Nomas\_beigu\_datums** (piemēram, 10.12.2020.)

#### Konstruktori:

Programma ietver sevī divus konstruktorus:

- **Public Noma()** {**Produkts\_pieejams** = **true**} – konstruktors nesaņem nekādus parametrus, bet pēc noklusējuma iestata, ka produkts ir pieejams.
- **Public Noma (string prod.\_kategorija, string prod.\_nosaukums, string tehn.\_raksturojumi)** { **Produkta\_kategorija** = **prod.\_kategorija**; **Produkta\_nosaukums** = **prod.\_nosaukums**; **Tehniskie\_raksturojumi** = **tehn.\_raksturojumi**; **Nomas\_cena\_dienā** = **nomas\_cena**; **Produkts\_pieejams** = **true**; } – konstruktors piedāvā savādīt visu nepieciešamo informāciju par produktu un iestata, ka produkts ir pieejams.

#### Metodes:

- **Nomas\_atlikušais\_laiks** (metode atgriež atlikušo laiku līdz nomas beigām)
- **Cena\_kopā** (metode atgriež kopējo summu par nomu, ņemot vērā, uz cik dienām ir iznomāts produkts)

- **Produkta\_info** (metode atgriež kopēju informāciju par produktu)
  - **Produkta\_info\_print** (metode izprintē datus par produktu, izmantojot standarta bibliotēku teksta datnē, ekrānā vai kādā citā datnē, piemēram, .pdf, izmantojot papildus bibliotēku)
  - **Nomnieks\_info** (metode atgriež kopējo informāciju par nomnieku)
  - **Nomnieks\_info\_print** (metode izprintē datus par nomnieku, izmantojot standarta bibliotēku teksta datnē, ekrānā vai kādā citā datnē, piemēram, .pdf, izmantojot papildus bibliotēku)

#### Lietotāja saskarne:

Lietotāja saskarne sastāv no trim galvenajām sadaļām (var lietot *MenuStrip*, *Pogas*, *Paneļus*, *ToolStrip*, *TabControl* u. c.):

- Datu ievade (sadaļa, kurā ievada datus par produktu un nomnieku)
- Datu izvade (sadaļa, kurā var izvadīt informāciju par produktu, nomnieku)
- Datu printēšana (atskaišu izveide) (sadaļa, no kuras var izprintēt informāciju par produktu vai nomnieku atsevišķā teksta vai cita formāta datnē)

Programma piedāvā:

- ievadīt datus par nomas produktu;
- ievadīt datus par nomnieku;
- izvadīt informāciju uz ekrāna, izmantojot izveidotās metodes (atlikušais laiks, cena par nomu kopā, produkta *info*, nomnieka *info*);
- saglabāt datus atskaites veidā kādā no datnēm (piemēram, .txt standarta bibliotēka vai .pdf ārējā bibliotēka).

#### Padziļināti:

Programma glabā un ielasa datus no kādas datnes (piemēram, .txt), līdz ar to dati nepazūd pēc programmas aizvēršanas.