

Kontrolltöö 1 aines *Objektorienteeritud programmeerimine*

Arvutimängus on tegelased ja esemed. Esemed on esitatud failis järgmisel kujul:

```
Tavaline haldjarüü;50
Võlupüksid;100
Välguvastane pross;250
Võlukeep;1000
```

Igal real on eseme nimetus ja eseme punktide arv (eraldatud semikooloniga).

Tegelastel on erinev arv esemeid. Nimetame siin mängus olevaid tegelasi ja esemeid ühise nimetusena üksusteks.

Kontrolltöö seisneb esemeid, tegelasi ja mängu käsitleva programmi koostamises. Programm peab vastama alltoodud nõuetele (isegi kui need kummalised tunduvad).

Programm peab sisaldama liidest `Üksus`, klasse `Ese`, `Tegelane`, `Mäng` ning peaklassi. Peaklassis loetakse sisse esemete nimekiri, luuakse tegelased ja simuleeritakse mängu. Peaklassis testitakse ka erinevate isendimeetodite tööd. Kõikide klasside kõik isendiväljad peavad olema privaatsed.

1. (1 p.) Liides `Üksus` sisaldab

1. `int`-tüüpi parameetriteta meetodit `punktideArv`;
2. `String`-tüüpi parameetriteta meetodit `info`.

2. (2 p.) Klass `Ese` realiseerib liidese `Üksus`.

1. Klassis on privaatsed isendiväljad järgmise info jaoks: nimetus (`String`) ja punktide arv (`int`).
2. Klassis peab olema kahe parameetriga konstruktor, mille abil saab määrata nimetuse ja punktide arvu.
3. Meetod `punktideArv` tagastab punktide arvu.
4. Meetod `info` tagastab selle eseme nimetuse.

3. (4 p.) Klass `Tegelane` realiseerib liidese `Üksus`.

1. Klassis on privaatsed isendiväljad järgmise info jaoks: nimi (`String`) ja esemete nimekiri (`List<Ese>`).
2. Klassis peab olema ühe parameetriga konstruktor, mille abil saab määrata nime.
3. Äsjaloodud tegelasel ei ole ühtegi eset. Esemel lisamiseks peab olema meetod `lisaEse`, mis jätab argumendiks antud eseme meelde.
4. Meetod `punktideArv` tagastab tegelase esemete punktide arvude summa.
5. Meetod `info` tagastab tegelase info tekstina, näidates tegelase nime, esemete arvu ja punktide arvu.
6. Klassis peab olema ka meetod `väljastaEsemed`, kus väljastatakse ekraanile tegelase esemed nii, et iga ese on eraldi real.
7. Klass `Tegelane` realiseerib liidese `Comparable<Tegelane>`, kusjuures `compareTo` meetod realiseeritakse nii, et võrdlemine toimub esemete arvu alusel.

4. (4 p.) Klassis `Mäng` peab olema privaatne isendiväli fikseeritud pikkusega tegelaste nimekirja (`Tegelane[]`) jaoks. Klassis peab olema ühe parameetriga konstruktor, mille abil saab määrata tegelaste nimekirja.

1. Klassis on `Tegelane`-tüüpi parameetriteta meetod `suurimaEsemeteArvuga`, mis tagastab suurima esemete arvuga tegelase. Meetodis tegelased sorteeritakse vastavalt meetodis `compareTo` kirjeldatud järjekorrale ja sorteeritud massiivist võetakse suurima esemete arvuga tegelast.
2. Klassis on `Tegelane`-tüüpi parameetriteta meetod `suurimaPunktideArvuga`, mis tagastab suurima punktide arvuga tegelase.

5. (5 p.) Peaklass peab olema nimega `Peaklass`. Klassis peab olema staatiline avalik meetod `loeEsemes` tagastustüübiga `List<Ese>`, mis võtab argumendiks failinime ja tagastab selles failis olevad esemete andmed. Meetod võib visata erindi (st meetodi signatuuris võib olla `throws Exception`). Esemete faili formaat on ülalpool toodud. Esemete arv failis ei ole teada (programm peaks töötama suvalise arvu esemetega (sh ka tühja failiga)). Kui failist lugemist ei õnnestu programmeerida, siis kirjutatagu selles meetodis vastav list programmi sisse (vähendab tulemust 2 punkti võrra).

Peameetod võib visata erindi (st meetodi signatuuris võib olla `throws Exception`).

Peameetodis tehakse järgmised tegevused.

1. Rakendatakse vastavat staatilist meetodit, et lugeda failist `esemed.txt` esemete andmed.
2. Luuakse vähemalt 5 tegelast (nimed mõelge ise välja).
3. Kõikidest tegelastest tehakse `Tegelane[]`-tüüpi massiiv. (Massiivi võib ka enne tegelaste tegemist luua ja järjest täita.)
4. Iga tegelase jaoks genereeritakse juhuslik arv n vahemikust $[2,10]$, mis näitab selle tegelase esemete arvu. Iga tegelase jaoks valitakse juhuslikult n eset. Selleks tuleb kasutada `Collections.shuffle` meetodit. Antud meetod võtab argumendiks listi ning järjestab selle suvalises järjekorras. Esemete list järjestada iga tegelase jaoks uuesti ümber ning lisada tegelasele esimest n eset.
5. Tegelaste info ja esemed väljastatakse ekraanile (kasutades vastavaid meetodeid).
6. Luuakse mäng kasutades tegelaste massiivi.
7. Leitakse ja väljastatakse ekraanile suurima esemete arvuga tegelase info (kasutades vastavaid meetodeid).
8. Leitakse ja väljastatakse ekraanile suurima punktide arvuga tegelase info (kasutades vastavaid meetodeid).

Andmete fail on aadressil <http://kodu.ut.ee/~marinai/esemed.txt>. Salvestage see oma arvutisse. Fail on kodeeringus UTF-8.