

## Pirmais praktiskais darbs: Divpersonu spēles ar pilnu informāciju izstrāde

**Punktu skaits:** 15 punkti

### Uzdevums:

Šis darbs ļauj studentiem pielietot praksē iegūtās zināšanas par pārmeklēšanā sakņotu problēmu risināšanu, izstrādājot spēles programmatūru. Darba izpildei studentam ir jāizvēlas nulles summas divpersonu spēle ar pilnu informāciju. Uzdevuma izpildei nav atļauts izvēlēties spēles "Krustiņi-nullītes" un "Connect 4" (kā arī minēto spēļu paveidus). Ja studentam ir grūtības atrast piemērotu spēli, tad ir iespēja izmantot kādu no spēlēm, kas tiek apskatītas lekcijās, to padarot sarežģītāku, vai konsultēties ar mācībspēku. Students var brīvi izvēlēties programmēšanas valodu vai vidi programmatūras izstrādei.

Programmatūrā ir jānodrošina šādas iespējas lietotājam:

- izvēlēties, kurš uzsāk spēli: lietotājs vai dators;
- izpildīt gājienus un redzēt izmaiņas spēles laukumā pēc gājienu (gan lietotāja, gan datora) izpildes;
- uzsākt spēli atkārtoti pēc kārtējās spēles pabeigšanas.

Programmatūrai ir jānodrošina grafiskā lietotāja saskarne (komandrindiņas spēles netiks pieņemtas). Šajā gadījumā runa nav par sarežģītu, 3D grafisko saskarni, bet gan par vizuālu elementu tādu kā izvēlnes, pogas, teksta lauki, ikonas, saraksti, utt. izmantošanu.

Izstrādājot programmatūru, studentam obligāti ir jārealizē:

- spēles koka vai tā daļas ģenerēšana;
- heuristiskā novērtējuma funkcijas izstrāde, ja spēles realizācijai tiek izmantots daļēji pieejams spēles koks;
- viens no studiju kursā apskatītajiem algoritmiem (Minimaksa algoritms, Alfa-beta vai Pārlūkošana uz priekšu pār n-gājieniem).

Studentam ir jāiesniedz atskaite par izstrādāto programmatūru. Atskaitei ir jāsaturs:

- programmatūras darbības demonstrācijas piemērs vai lietotāja ceļvedis ar paskaidrojumiem;
- apraksts par izmantotajām datu struktūrām ar detalizētiem komentāriem, kas tieši tiek glabāts konkrētajā datu struktūrā;
- heuristiskā novērtējuma funkcijas izmantošanas gadījumā detalizēts apraksts un pamatojums izvēlētajai funkcijai;
- realizēto pamatalgoritmu (spēles koka ģenerēšana, heuristisko vērtējumu piešķiršana virsotnēm, spēles algoritma pielietojums, uzvaru nesošo ceļu atrašana) kods ar studenta sniegtajiem skaidrojumiem. Kods atskaitei ir jāpievieno tikai teksta veidā. Nav atļauts to pievienot attēlu veidā.

Viss programmatūras kods ir:

- jāpievieno atskaites pielikumā teksta veidā;
- jāpadara arī pieejams kādā publiskā vietnē (GitHub, Google, u.c.) un saite uz to ir jānorāda uz atskaites titullapas. Izmantojot norādīto saiti, mācībspēkam ir jābūt iespējai lejupielādēt studenta darbu bez papildu reģistrācijas un ierobežojumiem.

Atskaite par darbu ir jāiesniedz elektroniskajā formātā (.docx vai .pdf), izmantojot e-studiju kursā sadaļā "Praktiskie darbi" pieejamo aktivitāti "Pirmā praktiskā darba atskaites iesniegšana".

Studentam ir jāaizstāv izstrādātā programmatūra. Aizstāvēšanas laikā studentam ir jāatbild uz jautājumiem par sistēmas uzbūvi, algoritmu realizāciju un datu struktūru izmantošanu. Studentam pašam ir jārūpējas, lai demonstrācija noritētu

veiksmīgi, t.i., būtu pieejamas demonstrācijai nepieciešamās tehnoloģijas un vides. Informācija par pieteikšanos aizstāvēšanai semestra laikā tiks publicēta e-studiju kursā sadaļā “Praktiskie darbi”.

Tādējādi izstrādājot darbu, studentam ir jāizpilda šādi soļi:

1. jāizvēlas spēle;
2. jāizvēlas programmēšanas vide/valoda;
3. jāprojektē, jārealizē un jātestē spēle;
4. jā sagatavo atskaite par izstrādāto spēli un tā ir jāiesniedz e-studiju kursā;
5. jāvienojas par aizstāvēšanas laiku.

### **Termiņi:**

E-studiju kursa sadaļā “Praktiskie darbi” ir publicēti gala termiņi atskaites nodošanai par izstrādāto programmatūru. Pēc norādītajiem termiņiem atskaites vairs netiks pieņemtas un programmatūru vairs nebūs iespējams aizstāvēt.

### **Plaģiātisma pārbaude**

Studentam ir jāņem vērā, ka izstrādātajai atskaitai un kodam tiks veikta plaģiātisma pārbaude. Līdz ar to jebkuri mēģinājumi (pilnībā vai daļēji) kopēt citu studentu/personu radīto vai tīmekļa vietnēs vai jebkurā citā informācijas avotā pieejamo kodu tiks uzskatīti par plaģiātisma gadījumu, kura rezultātā tiks sastādīts ziņojums par studējošā akadēmiskā godīguma pārkāpumu un tiks iesniegts augstskolas administrācijai, kā arī tiks anulēti studenta punkti par praktisko darbu. Punktu anulēšana un ziņojuma sastādīšana var notikt arī vēlāk, ja pārkāpums tiks konstatēts jau pēc darba aizstāvēšanas.

### **Vērtēšanas kritēriji**

Vērtējot darbu, mācībspēks ņems vērā:

- atskaites kvalitāti (vai ir iekļauta iepriekš prasītā informācija un vai tā ir jēgpilna);
- izstrādātās programmatūras kvalitāti un atbilstību prasībām;
- sniegtu atbilžu kvalitāti aizstāvēšanas laikā;
- akadēmiskā godīguma pārkāpuma neesamību.

### **Darbs automātiski netiks izskatīts un aizstāvēšana netiks nozīmēta, ja:**

- students nav iesniedzis darba atskaiti, bet piesakās darba aizstāvēšanai;
- students iesniedza komandrindiņas programmatūru;
- studenta darbā spēles koks netiek ģenerēts, bet ir predefinēts vai glabāts failā;
- studenta darbā netiek realizēts studiju kursā apskatītais algoritms (Minimaksa algoritms, Alfa-beta vai Pārlūkošana uz priekšu pār n-gājieniem), bet dators izpilda gājienus gadījuma ceļā, vai gājieni ir predefinēti kodā, u.c.;
- ir konstatēts akadēmiskā godīguma pārkāpums;
- students uz atskaites titullapas nav norādījis saiti uz kodu publiskajā vietnē;
- atskaitē students pievienoja kodu attēlu veidā.