Rühmatöö kirjeldus: Ühenda 4

* autorite nimed;

Sandra Lannes ja Sarah Lannes

* projekti põhjalik kirjeldus, kus on kirjas programmi eesmärk ja selgitus programmi üldisest tööst, vajadusel lühike kasutusjuhis;

Meie tegime oma projektiks populaarse lauamängu Ühenda 4 (Connect Four). Mängu eesmärgiks on olla esimene, kes suudab paigutada 4 oma märki ühte ritta horisontaalselt, vertikaalselt või diagonaalselt. Mäng arendab loogilist mõtlemist, keskendumist, jälgimist, täpsust ja kannatlikkust. Märke saab paigutada nii, et need kukutatakse ülevalt alla üksteise peale(vaata pilti). 

* iga klassi kohta eraldi selle eesmärk ja olulisemad meetodid;

Main- peaklass; psvm- meetod, milles loome Mangijate isendid ning käivitame mängu

Mangija- klass kuhu salvestame Mangijate(mängijate) isendid; konstruktor ning getMangija - meetodid, mille abil loome Mangija isendid ning väljastame neid hiljem ekraanile.

Mang- klass, kus kogu mäng toimib; kaik - meetod, milles mängija märk asetatakse mängulauale, kasNeliReas - meetod, mille abil kontrollime, kas neli samasugust märki on mõnes reas, veerus või diagonaalis, parisMang - meetod, kus tehakse kõik vajalikud tegevused, et mäng toimiks.

* projekti tegemise protsessi kirjeldus (erinevad etapid ja rühmaliikmete osalemine neis);

Tegime projekti suht ühe hooga ära, erinevaid etappe ei oskagi väga välja tuua, võtsime suhtumise et “hakkame tegema ja vaatame mis siis saama hakkab”. Ideed arutasime üheskoos ning koodi kirjutasime ka üsna paralleelselt, jagamiseks kasutasime githubi. Sandra tegi põhja, mõned väiksemad meetodid ning kontrollimismeetodi, Sarah täiendas põhja ning tegi käikude käimise süsteemi ning pani lõpuks mängu meetoditest kokku. Testisime ja kommenteerisime koodi jooksvalt töö käigus (igaüks enda osa) ning lõpliku kirjelduse kirjutasime google docsis dokumenti jagades.

* iga rühmaliikme panus (sh tehtud klassid/meetodid) ja ajakulu (orienteeruvalt);

Sandra tegi meetodid valiAlustaja, kasNeliReas, kontrolliDiagonaalid ja põhja.

Sarah tegi meetodid taidaTabel, käik, valjastaTabel, ja parisMang ja aitas ka ülejäänuga.

Projekti ajakulu kokku oli ilmselt umbes 10h.

* tegemise mured (nt millistest teadmistest/oskustest tundsite projekti tegemisel puudust);

Kindlasti oli kõige raskem teha meetodit diagonaalide kontrollimise jaoks.Selleks otsisime abi googlest ning üritasime välja mõelda sobivat algoritmi, mis kontrolliks ilusti kõiki diagonaale. Samuti valmistas scanner klassi kasutamine esialgu segadust, kuid see kadus kiirelt. Samuti valmistas veidi raskusi ühtse vormistuse ja stiili leidmine, kuna meil on välja kujunenud veidi erinev stiil.

* hinnang oma töö lõpptulemusele (millega saite hästi hakkama ja mis vajab arendamist);

Oleme tööga rahul, kuna saime kõik toimima nii kuidas soovisime. Arendamist vajaks kindlasti kasutajaliides, kuna reaalset pilti nähes oleks kindlasti palju huvitavam mängida.

* selgitus ja/või näited, kuidas programmi osi eraldi ja programmi tervikuna testisite ehk kuidas veendusite, et programm töötab korrektselt.

Pärast iga meetodi loomist katsetasime selle õigesti toimimist peaklassis (andsime ette erinevaid väärtusi, proovisime programmi “katki” teha). Lõpuks tervikuna testimiseks palusime oma pereliikmetel sellega mängida, et leida üles ka vead, mis ei pruugi olla koodis (mängu kehv arusaadavus, jne).