

ROBOOTIKA E-ÜLESANNE

PUNKTID 8. KLASS

Koht	Meeskonna nimi	Protsess on õigesti kirjeldatud	Võimalikult vähe elemente	Õiged tähised ja tingmärgid	Visuaalselt selge	Kokku ülesande punktid	Trahvi-punktid ületatud aja eest	Koond-tulemus
	<i>Maksimaalne tulemus</i>	50	30	15	5	100		100
1.	Tervislik toidupüramiid	50	30	15	5	100		100
2.	Jalo	49	30	15	5	99		99
3.	Lando	40	30	15	5	90		90
4.	Tähenärijad	50	22	15	5	92	-3	89
5.	Millimeetri Mehed	45	0	15	5	65		65

TAGASISIDE MEESKONDADELE:

Tervislik Toidupüramiid

100p – Oma vanusegrupi parim töö. Meeldib, et on arvestatud ka sellega, et me ei tea mis asendis masin on, kui me ta enda kätte saame, kas tuli juba põleb või mitte.

Jalo

99p – Sisuliselt praktiliselt ideaalne. Ainukene asi, mida saab juurde teha, on küsida alguses, kas tuli juba põleb. Olenevalt süsteemist võib juhtuda mitte midagi, kui üritame lülitada sisse asja, mis on juba sisse lülitatud (lambipirn) või võime teha süsteemile niimoodi kahju (töötava auto mootori "uuesti käivitamine" kuigi ta juba töötab.

Lando

90p – On ebaselge kas me peaksime algselt mööda UML-i liikuma alla v paremale.

Millimeetri mehed

65p – antud koodi saab teha kordades lühemalt. Kui me tahaks seda koodi muuta niimoodi, et tuli süttib näiteks 75 korda, tuleks kood 15 korda pikem, kui niimoodi teha. Tegelikult saab teha sama pika koodi, sõltumata sellest, kas me käivitame tule 2 korda või 2 miljonit korda.

Tähenärijad

92p - Põhimõte on hea, kuid kood on ebaefektiivne. Kustutate tule ära ja siis järgmise asjana küsite "kas tuli on kustutatud." Pole vaja selles kohas küsida, ise ju lülitasite just selle tule välja.



PUNKTID 9.-10. KLASS

Koht	Meeskonna nimi	Protsess on õigesti kirjeldatud	Võimalikult vähe elemente	Õiged tähised ja tingmärgid	Visuaalselt selge	Kokku ülesande punktid	Trahvi-punktid ületatud aja eest	Koond-tulemus
	<i>Maksimaalne tulemus</i>	50	30	15	5	100		100
1.	Igavesti hõbedas	50	30	15	5	100		100
2.	Punkt	47	30	15	5	97		97
3.	Bonkers	47	30	15	5	97	-1	96
4.	Ässad nr 1	45	30	15	5	95		95
5.	Konn ja bande	47	27	15	5	94		94
6.	P1	50	25	15	3	93		93
7.	Phronesis	47	25	15	5	92		92
8.	Sputnik	50	20	15	5	90		90
9.	Paradox	38	30	15	5	88		88
10.	Vaprad ja ilusad	35	25	15	5	80		80
11.	Teadusuurijad	50	0	10	5	65		65
12.	Rakettlased	10	15	15	0	40		40
13.	Eukalüpt	0	20	0	5	25		25
14.	Merka	0	0	15	5	20		20
-	Mutahavir	0	0	0	0	0	-2	0
-	Viimsi Virmalised	0	0	0	0	0	-1	0

TAGASISIDE MEESKONDADELE:

P1

Ülesannet saab lühemalt teha, kui rakendada eraldi X ja Y muutujate asemel muutujat I. See säästab mälu mahtu. Saab teha eraldi situatsioonid $I < 5$ ja $I < 10$. Muidu sisuliselt ideaalne töö, visuaalselt on natukene keeruline jälgida Y loop tsükli, mitu joont saavad üksteise ligidal kokku.

Ässad nr 1

Ideaalsest tööst lahutab üks asi: antud diagramm ei selgita kuidas me teame mitmes ring meil käsil on. Seda saab lahendada näiteks läbi I ja I++.

Vaprad ja ilusad

Punane tuli süttib 4 korda, kuigi peaks süttima ainult 1 korra. Pole selge kuidas programm peab arvet selle üle mitu korda tuli juba süttinud. Seda saab lahendada läbi I ja I++. Igas loopis on üks üleliigne otsuse koht, millele ei ole mingit väärtust antud.

Sputnik

Koodis on 4 tule ära kustutamise blokki, kuigi ülesandes on ainult kaks tule ära kustutamise käsku. Koodi on võimalik optimeerida niimoodi, et nelja bloki asemel oleks ainult kaks (kustuta roheline, kustuta kollane), kui tõsta küsimise küsimise blokk ümber. Veidi koodi struktuuri modifitseerides on võimalik ka omada ainult üht blokki (kustuta mistahes tuli).

Paradox

Meeldib, et on kasutatud muutujat. Punane läheb põlema 5 korda, peaks minema põlema ainult ühe korra. Väga meeldib, et on kasutatud värve süsteemi iseloomustamiseks. Koodi parempoolse osa saab tõsta koodi vasakpoolse osa sisse, tehes koodi "üheharaliseks" ja kergemini jälgitavaks. Hetkel peab mõlemad poolt samaaegselt käsitlema ning mõtlema süsteemi taktkiiruse peale ning see võib tekitada probleeme.

Konn ja bande

Pole selge kuidas programm peab arvet selle üle mitu korda tuli juba süttinud. Seda saab lahendada läbi I ja I++. Ebaselge on ka mille järgi programmi lõppedes peaks otsustama kumb lõppsündmus tuleb, kas Y või N.

Phronesis

Teie muudatus on väga oluline ja toob teile 50 lisapunkti. Selgusetu on kuidas me teame mitu korda tuli on põlema läinud (I ja I++) ning pärast mõlemat 5 tule põlemise tsükli on üks üleliigne teadvustus (X värvi tuli on nüüd põlenud 5 korda).

Teaduseuurijad

Kood on ebamõistlikult pikk. Mis te teeksite, kui ülesanne oleks 5000 korda panna roheline põlema ja siis 5000 korda panna kollane põlema? Ei ole mõistlik kõiki neid 10000 korda välja kirjutada. Seda situatsiooni saab lahendada muutujaga I, mis loendab mitu korda sündmust X on toimunud. Puudu on programmi algust ja lõppu tähistav märgistus.

Eukalüpt

Tegemist ei ole UML koodiga. "Wait 1 second" on üleliigne.

Rakettlased 2.0

Keeruline on aru saada mis peaks juhtuma programmi käivitades. Kas pannakse tuli põlema, liigutakse signaali bloki juurde või küsitakse kas $n=5$? Küsitakse n väärtust, kuid kuskil pole kirjeldatud mis n väärtus on või kuidas see muutub.

Punkt

Kõik on õige peale selle, et meil ei ole mingit moodust, kuidas loendada kui mitu korda üks või teine tuli on põlema läinud. Selle jaoks tuleks kasutada näiteks muutujat I ja I++.

Merka

On näha, et olete üritanud kasutada UML koodi, kuid on võimatu aru saada, mida antud kood üritab saavutada. Blokkide juures ei ole ühtegi kirjeldust, näiteks, kas esimene küsimus on "kas roheline tuli põleb" või "kas kolm on suurem või seitse" või "kas täna on esmaspäev."

Igavesti hõbedas

Teie tähelepanek on õige. Hindasin 9.-10. klassi tööd, kuna see on see, mis oli Teile mõeldud. Kõige parem töö. Kõik on ideaalne. On kasutatud muutujat, värvid on selgelt eristuvad, visuaalselt on kerge jälgida, mis toimub. Töös, mida ei hinnanud, oleksite saanud kõrged (aga mitte max) punktid.

Bonkers

Kood on kergesti jälgitav. Arusaamatu on kuidas me teame mitu korda mõni tuli on süttinud. Seda saab lahendada läbi I ja I++.

Viimsi Virmalised

On näha, et ülesanne oli teie jaoks liiga keeruline. Loodetavasti järgmiste teemade ülesanded on rohkem teemakohased.

Mutahavir

On näha, et ülesanne oli teie jaoks liiga keeruline. Antud ülesannet võis lahendada ka paberi ja pliiatsiga. Loodetavasti järgmiste teemade ülesanded on rohkem teemakohased.



PUNKTID 11.-12. KLASS

Koht	Meeskonna nimi	Protsess on õigesti kirjeldatud	Võimalikult vähe elemente	Õiged tähised ja tingmärgid	Visuaalselt selge	Kokku ülesande punktid	Trahvi-punktid ületatud aja eest	Koond-tulemus
	Maksimaalne tulemus	50	30	15	5	100		100
1.	Veenilaiend	50	30	15	5	100		100
2.	Team A	50	0	15	5	70		70
3.	Timm	15	30	15	5	65		65
4.	Täheke	10	30	15	5	60	-2	58
5.	Meesinep	0	10	15	5	30		30
6.	Võsuf est	0	0	0	5	5		5

TAGASISIDE MEESKONDADELE:

Meesinep

30p – Diagramm ei kirjelda toimuvat protsessi korrektselt. Roheline ja kollane tuli süttivad ning 3. korral süttib ka sinine tuli, koos rohelise ja kollase tulega. Lisaks sellele 6. ja 9. korral ei sütti sinine tuli.

Timm

65p – Diagrammilt ei tule välja kuidas programm peaks teadma mitmes ringi käsil on või kust kohast uus ring algab. Puudu on ringiarvude üle arvet pidav muutuja. Lisaks sellele süttib igal ringil 10 tuld (5 rohelist, 5 kollast) aga peaks süttima üks tuli, kas roheline (1.-5. ringi ajal) või kollane (6.-10. ringi ajal), v.a. sinised ringid, mis on õigesti sisse arvestatud.

Team A

70p – Diagramm kirjeldab toimuvat protsessi õigesti, kuid on ebamõistlikult pikk. Kui Teil palutaks teha sama program, ainult miljoni kordusega (näiteks valgusfoor), siis te ei hakkaks kirjutama miljonit koodirida, et mida valgusfoor teeb 776257. korral, mida 776258. korral, mida 776259. korral jne. Samamoodi, kui tuleks ülesanne, et iga neljas on sinine, mitte iga kolmas, peaksite kõik miljon rida ümber kirjutama. Ebamõistlik. Saab teha oluliselt lühemalt, loendades, kas hetkel on 3-ga jaguv kordade arv.

Võsuf est

5p - Tegemist ei ole UML koodiga. Samuti pole antud koodis näha erinevate värvide kasutamist. Antud lahenduse peale vaadates ei ole võimalik teha järeldust, et koodi kirjutaja on soovinud saavutada olukorda, mis talle on ülesandena ette antud. Kood on kergesti jälgitav, autor on kasutanud treppimist ja antud koodi mõistes mõistlikke tähiseid ja tingmärke.

Veenilaiend

100p – väga meeldib, et olete kasutanud muutujaid. Tegelikult piisab ühest muutujast, ei pea kasutama eraldi R ja K muutujat, saab mälu ja keerukust kokku hoida ja vähem defineerida, kui kasutada vaid ühte muutujat I, ning defineerida ära programmi käitumine olukorras, kus $I < 5$ ja olukorras, kus $I < 10$. Kui nüüd tõsta $I \% 3 = 0$ kõige ette, siis ei pea teda kaks korda kasutama, ühe kora mõlemas tsükli, vaid piisab, kui teda 1 korda kasutada kõige ees ja seejärel otsustada kas käivitada $I < 5$ kontroll (ja vajadusel $I < 10$



Rakett 21

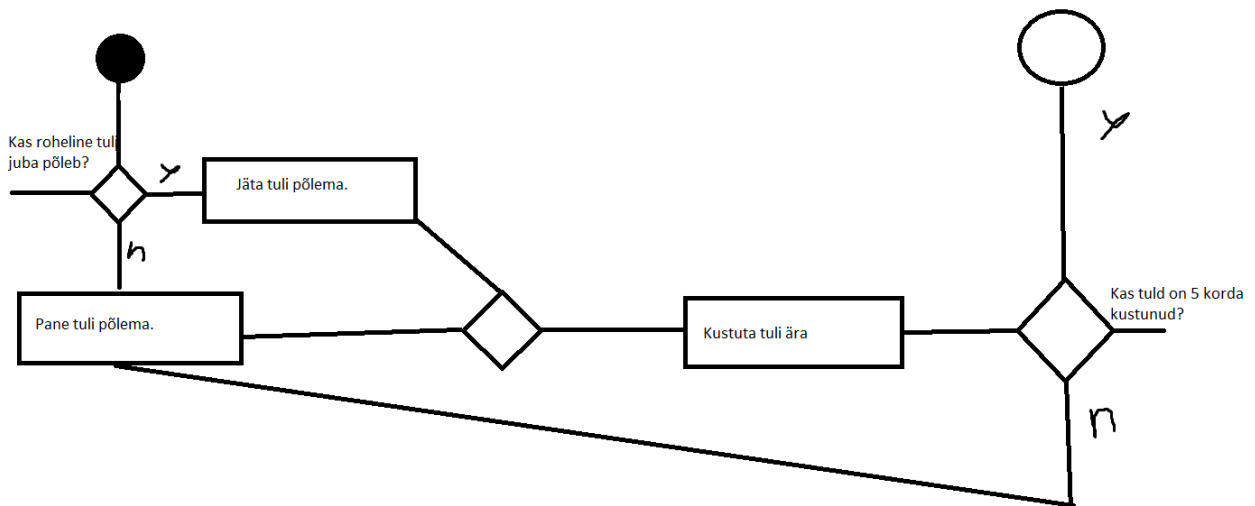
kontroll) või minna kohe sinise tule juurde. Olete ainukene võistkond oma vanuseklassis, kes on kirjeldanud muutuja kasutamist.

Täheke

60p - 3., 6. Ja 9. korral ei sütti sinine tuli. Sinine tuli süttib ainult 16. Korral, pärast viit rohelist, viit kollast ja viit punast. Samuti on kirjeldamata kuidas programm teab mitu korda tuli süttinud on.

LAHENDUSED:

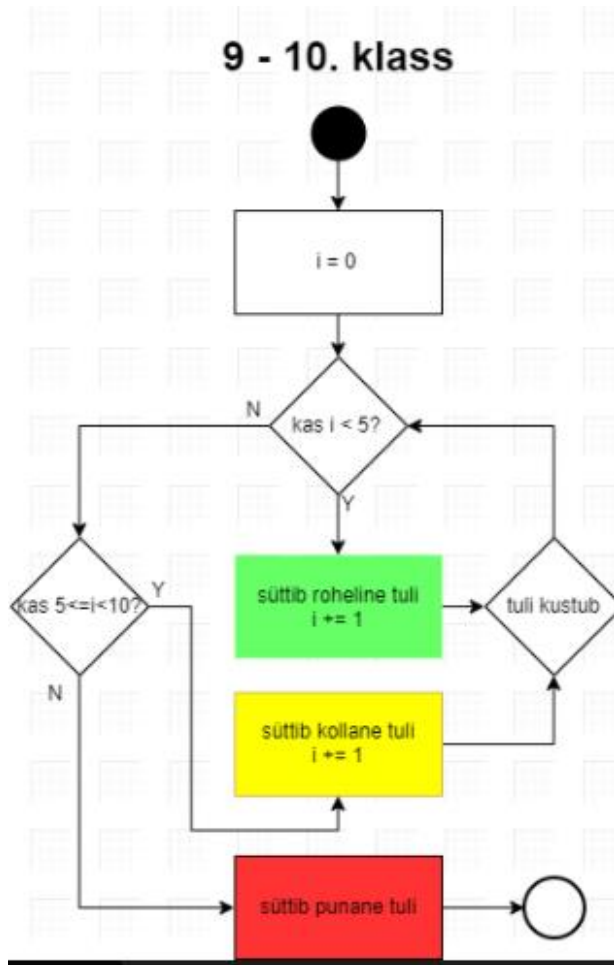
8. klassi parim lahendus



Peter William Stokes, Uku Sinkkonen, Sander Kottel, Holger Paulus



9.-10. klassi parim lahendus



11.-12. klassi parim lahendus

