***გამოცდის ფორმატი***

**\*მონიშნეთ გამოცდის ფორმატი (მიუთითეთ √)**

|  |  |
| --- | --- |
| დახურული წიგნი |  |
| ღია წიგნი | **√** |
| **\*ღია წიგნის შემთხვევაში მონიშნეთ გამოცდაზე ნებადართული ელემენტები (მიუთითეთ √)** | |
| სალექციო მასალები (პრეზენტაცია და სხვა) |  |
| ელექტრონული წიგნები | **√** |
| წიგნები |  |
| კონსპექტები |  |
| ლექსიკონი |  |
| კალკულატორი |  |
| ლეპტოპი/პლანშეტი |  |

**\* გამოცდის ჩატარების წესი იხილეთ ,,დესკტოპზე“ საქაღალდეში Exam materials**

***საგამოცდო საკითხების ფორმა***

***ვარიანტი # 4***

|  |  |  |  |
| --- | --- | --- | --- |
| ***სკოლა*/*საგანმანათლებლო პროგრამა*** | მათემატიკა და კომპიუტერული მეცნიერება | ***სტუდენტის მიერ მიღებული ქულა*** |  |
| ***საგანი*** | პროგრამირების პარადიგმები | | |
| ***ლექტორი*** | შ. ღვინეფაძე | | |
| ***კურსი*** | I | | |
| ***ჯგუფი*** |  | | |
| ***გამოცდის ფორმა*** | ღია წიგნი | | |
| ***გამოცდის***  ***ხანგრძლივობა*** | 2 საათი | | |
| ***მაქსიმალური ქულა*** | 120 | | |
| ***სტუდენტის სახელი და გვარი:*** | | | |

**ნაშრომის ჩაბარების წესი:**

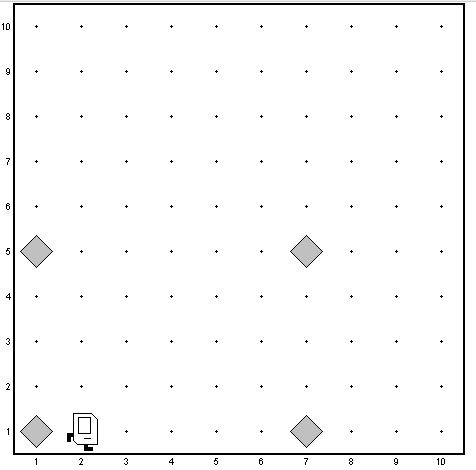
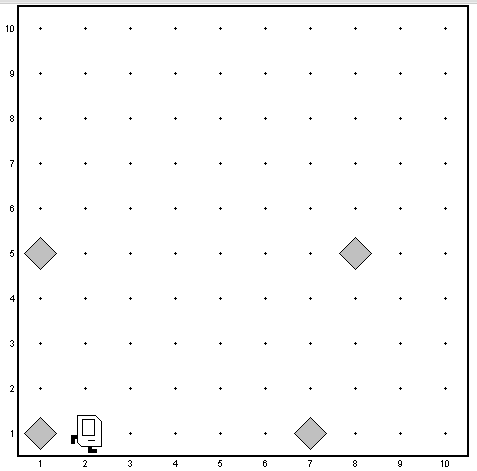
**შექმენით დესკტოპზე თქვენი მეილის პრეფიქსის სახელის მქონე java გაფართოების ფაილი(მაგალითად gboch16.java). გახსენით ეს ფაილი notepad++-ის საშუალებით. ამისათვის შეგიძლიათ ფაილზე დააკლიკოთ მაუსის მარჯვენა ღილაკით და მონიშნოთ open with notepad++. ფაილის დასაწყისში დაწერეთ თქვენი სახელი და გვარი და ვარიანტის ნომერი. ყველა ამოცანის ამოხსნა ამ ფაილში უნდა დაწეროთ. გამოყავით ამოცანები ნომრებით 1,2,3...  
გამოცდიდან გასვლის წინ აუცილებლად გადაამოწმეთ რომ თქვენი ფაილი არის დესკტოპზე.**

**წარმატებები ;)**

**ამოცანა 1. კარელი (20 ქულა)**

მოცემული გაქვთ სამყარო, კარელი დგას მარცხენა ქვედა კუთხეში სახით მარჯვნივ და ჩანთაში აქვს უსასრულო რაოდენობის ბრილიანტები. ცნობილია რომ მხოლოდ (1, 1) უჯრაზე და კიდევ სამ სხვა უჯრაზე დევს თითო ბრილიანტი. კარელმა უნდა გაარკვიოს, ბრილიანტებით მონიშნული წერტილები არის თუ არა მართკუთხედის წვეროები, თუ არის მაშინ (1, 1) უჯრაზე დადოს კიდევ ერთი ბრილიანტი, თუ არა და პირიქით აიღოს (1, 1) უჯრიდან ბრილიანტი.

მაგალითისთვის იხილეთ ქვემოთ მოცემული სურათები, მარცხენა სურათში ბრილიანტები მდებარეობენ მართკუთხედის წვეროებზე, ხოლო მარჯვენაში არა.

კარელი სად იდგება მოქმედების შესრულების შემდეგ არ აქვს მნიშნელობა.

**ამოცანა 2. რას დაბეჭდავს (20 ქულა)**

დაწერეთ რას დაბეჭდავს შემდეგი პროგრამა. თუკი შეცდომა მოხდება კომპილაციის ან გაშვების შემდეგ მიუთითეთ რა სახის შეცდომაა.

**public** **class** Problem2 **extends** ConsoleProgram{

**public** **void** run(){

**int** x = 2;

**int** y = 4;

GOval o = **new** GOval(4,2); //width, height

println(racxa(x,2,o));

println(racxa(y,x/2+x,o));

}

**public** **int** racxa(**int** x, **int** y, GOval o){

String s = "hello world";

x = varesiRacxa(x/y, x+(**int**)o.getHeight(), s);

o.scale(2);

y = varesiRacxa(x, y+(**int**)o.getHeight(), s);

**return** x+y;

}

**public** **int** varesiRacxa(**double** x, **int** y, String s){

**int** z = (**int**)x + s.charAt(0)-s.charAt(1);

s = s.substring(1);

**return** z + y;

}

}

**ამოცანა 3. გეომეტრიული პროგრესია (20 ქულა)**

თქვენი ამოცანაა დაადგინოთ არის თუ არა მომხმარებლის მიერ შემოყვანილი რიცხვები გეომეტრიული პროგრესია. ამისათვის თქვენ უნდა დაწეროთ კონსოლის პროგრამა.

ჩათვალეთ რომ მომხმარებელს მხოლოდ მთელი რიცხვები შემოჰყავს მანამ სანამ არ შემოიყვანს -1. როგორც კი მომხმარებელი -1 შემოიყვანს თქვენმა პროგრამამ უნდა დაბეჭდოს სტრინგი “Geometric progresson” თუკი შეყვანილი მიმდევრობა არითმეტიკული პროგრესიია და უნდა დაბეჭდოს სტრინგი “Not a geometric porgression” წინააღმდეგ შემთხვევაში. რის შემდეგაც პროგრამამ უნდა დაასრულოს მუშაობა

**public** **class** GeometricProgression **extends** ConsoleProgram{

**public** **void** run (){

}

}

**ამოცანა 4. ენპლეტი (25 ქულა)**

ტექსტში ენპლეტი ეწოდება n ცალ ერთმანეთის მიყოლებით მდგარ ერთიდაიგივე სიმბოლოებისგან შემდგარ სტრინგს. თქვენი ამოცანაა ტექსტში წაშალოთ ყველა შემხვედრი ენპლეტი. ამისათვის უნდა დაწეროთ მეთოდი removeNplet() რომელსაც გადაეცემა სტრინგი და მთელი რიცხვი n და რომელიც აბრუნებს სტრინგს. მეთოდმა უნდა იმოქმედოს შემდეგნაირად: უნდა წაშალოთ პირველივე(ყველაზე მარცხნივ მდგომი) ენპლეტი, ამის შემდეგ თუ მიღებულ სტრინგში ენპლეტი აღარ არის დააბრუნოთ ეს სტრინგი თუკი არის მაშინ ისევ წაშალოთ პირველივე ენპლეტი და ა.შ. მაგალითად,

removeNplet(“asdXXXasd”,3) -> “asdasd”

removeNplet(“asdzzdas”,2) -> “asas”

**private** **String** removeNplet(**String** text, **int** n){

}

**ამოცანა 5. 3-ჯერ კლიკი (40 ქულა)**

დაწერეთ გრაფიკული თამაში, რომელიც გაშვებისთანავე დაამატებს CIRCLE\_NUM ცალ CIRCLE\_D დიამეტრის მქონე წრეს შემთხვევით ლოკაციებზე.

ამის შემდეგ პროგრამა ყოველ DELAY მილიწამში ერთხელ შემთხვევით ადგილზე უნდა ამატებდეს ახალ წრეს, მანამ სანამ თამაში არ დასრულდება. თუკი მოთამაშე 3 ჯერ ზედიზედ დააკლიკებს ერთსა და იმავე წრეს, პროგრამა უნდა შლიდეს წრეს კანვასიდან. გაითვალისწინეთ, რომ წრე არ წაიშლება თუ 3 ჯერ ზედიზედ არ იქნა მასზე დაკლიკებული. ანუ ამ 3 კლიკს შორის არ უნდა მოხდეს კლიკი სხვა წრეზე ან კლიკი ცარიელ ადგილზე. თამაში მთავრდება როცა ყველა წრე გაქრება დაფიდან. თამაშის დამთავრებისას პროგრამაც უნდა გაჩერდეს და აღარ დახატოს ახალი წრეები.

შენიშვნა: წრეების დამატება უნდა მოხდეს ფანჯრის საზღვრებს შიგნით.

**public** **class** Green **extends** GraphicsProgram{

**private** **final** **static** **int** ***CIRCLE\_D*** = 40;

**private** **final** **static** **int** ***CIRCLE\_NUM*** = 50;

**private** **final** **static** **int** ***DELAY*** = 2000;

**public** **void** run(){

}

}

**ამოცანა 6**. ბონუსი **(20 ქულა)**

თუკი ბონუსის მიღების სურვილით ხართ შეპყრობილი აირჩიეთ და დაწერეთ მთელი რიცხვი 0-დან 100-მდე შუალედში(0-ის და 100-ის ჩათვლით). გამოცდის შემდეგ დავითვლით თითოეული თქვენგანის მიერ არჩეული რიცხვების საშუალო არითმეტიკულს, გავამრავლებთ -ზე და ავიღებთ მიღებული რიცხვის მთელ ნაწილს. ვისი არჩეული რიცხვიც დაემთხვევა ჩვენს მიერ გამოთვლილ რიცხვს ის მიიღებს ბონუს 20 ქულას.