***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 # 1***

|  |  |  |  |
| --- | --- | --- | --- |
| ***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:*** | | | |

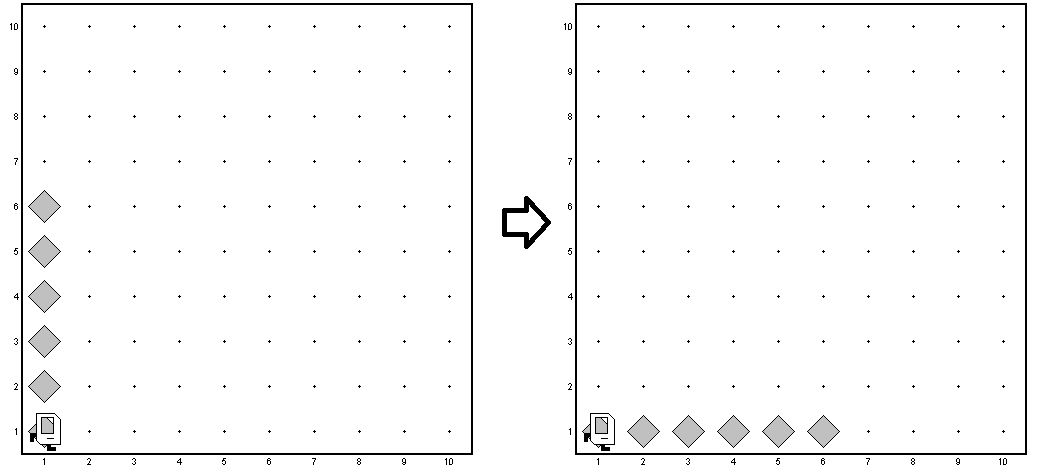
**ნაშრომის ჩაბარების წესი:**

**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 კარელის სამყარო სადაც პირველ სვეტში (1, 1) უჯრიდან (1, k) უჯრამდე განლაგებულია ბრილიანტები, უჯრაში თითო. კარელი დგას მარცხენა ქვედა კუთხეში, სახით მარჯვნივ და ჩანთაში აქვს უსასრულო რაოდენობის ბრილიანტები. თქვენი ამოცანაა კარელს გადაალაგებინოთ ბრლიანტები პირველი ვერტიკალიდან, პირველ ჰორიზონტალში (1, 1) უჯრიდან (k, 1) უჯრამდე, როგორც სურათზეა ნაჩვენები.



ჩათვალეთ რომ სამყაროს ყოველთვის აქვს ვერტიკალში არსებული ბრილიანტების ჰორიზონტალურად განლაგებისთვის საკმარისი სიგანე. 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 = 3;

GOval o = **new** GOval(4,3); //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. ლოგარითმი (20 ქულა)**

თქვენი ამოცანაა დაწეროთ მეთოდი რომელიც ლოგარითმს გამოითვლის. შეგახსენებთ რომ ლოგარითმი n-ის ფუძით m არის რიცხვი რომელ ხარისხშიც უნდა ავიყვანოთ n იმისათვის რომ m მივიღოთ. ანუ, თუ

, მაშინ

დაწერეთ მეთოდი log რომელსაც გადაეცემა 2 მთელი რიცხვი და რომელიც აბრუნებს ამ რიცხვებითვის ლოგარითმის მთელ ნაწილს.

თქვენ არ გავთ უფლება გამოიყენოთ Math კლასი. შესაბამისად ვერ გამოიყენებთ მეთოდებს log, log10, log1p, pow და ა.შ.

ჩათვალეთ, რომ m>n>0.

**private** **int** log(**int** n, **int** m){

}

**ამოცანა 4. ტექსტის შეკუმშვა (25 ქულა)**

მოცემული გაქვთ ტექსტის შეკუმშვის ალგორითმი.

თუკი ტექსტში რომელიმე სიმბოლო ზედიზედ ერთზე მეტჯერ გვხვდება ერთმანეთის გვერდით, მაშინ შეგვიძლია ეს ტექსტი შევამციროთ და ამ სიმბოლოების მაგივრად დავწეროთ თუ რამდენჯერ მეორდება და რა სიმბოლოა. მაგალითად, “aaaaa”-ს მაგივრად შეიძლება დავწეროთ “5a”.

თქვენი ამოცანაა დაწეროთ მეთოდი longestRunCompression(), რომელსაც გადაეცემა მხოლოდ პატარა ლათინური ასოებისგან შემდგარი სტრინგი და აბრუნებს სტრინგს. გადაცემული სტრინგი ტექსტია, რომლის შეკუმშვაც გვსურს. მეთოდმა უნდა დააბრუნოს შეკუმშული ტექსტი.

მაგალითად

longestRunCompress(“aabbbbcc”) -> “2a4b2c”

longestRunCompress(“abbbbcccdee”) -> “a4b3cd2e”

**private** **String** longestRunCompression(**String** text){

}

**ამოცანა 5. ოხერი წრე (35 ქულა)**

თქვენი ამოცანაა დაწეროთ გრაფიკული პროგრამა, რომელიც ფანჯრის ცენტრში დახატავს CIRCLE\_D დიამეტრის მქონე შავად გაფერადებულ წრეს.

გარდა ამისა პროგრამა უნდა აკმაყოფილებდეს ორ მოთხოვნას:

• მაუსის ნებისმიერ ადგილზე კლიკით, წრე უნდა იცვლიდეს ფერს რომელიმე შემთხვევით ფერზე.

• მაუსის დრაგზე, თუ დარგი მოხდა ცენტრისკენ წრე უნდა შემცირდეს შესაბამისი მანძილით და პირიქით წრე უნდა გაიზარდოს თუ დრაგი მოხდა ცენტრიდან ფანჯრის საზღვრებისკენ. ანუ თუ დრაგის დროს მაუსი წრის ცენტრს უახლოვდება, რამდენი პიქსელითაც უფრო ახლოს არის მაუსი ცენტრთან იმდენი პიქსელით უნდა შემცირდეს წრეწირის რადიუსი. თუ მაუსი წრეწირის ცენტრს შორდება, მაშინ წრის რადიუსი უნდა გაიზარდოს იგივე პრინციპით.

**public** **class** Circle **extends** GraphicsProgram{

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

**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.