

მოდელირება და ვიზუალიზაცია

საკონტროლო 1

5 მაი. 2026 წ.

1 წერტილი მართკუთხედში

10 ქულა

გადაიწერეთ არქივი `modeling_02.zip`. `randpoints` პროექტში გვაქვს 2დ წერტილის კლასი `utility.h` განმარტებულია შემთხვევითი `float` ტიპის რიცხვების გენერატორი რაღაც მოცემულ შუალედში. პროექტში მოცემულია ამ კლასის და ფუნქციის გამოყენების მაგალითი. გამოიყენეთ ეს კლასი და ეს ფუნქცია.

`pointinbox` პროექტში უკვე დამატებულია ზემოთ ნახსენები კლასი და ფუნქციები. ამ პროექტში გაკეთებულია მართკუთხედის და წერტილის დახატვა. მართკუთხედი მოცემულია როგორც მარცხენა დაბალი წერტილი, სიგანე და სიმაღლე. იხ. კოდი ქვემოთ

კოდი 1.1. წერტილის და მართკუთხედის დახატვა

```
1 // use proper includes for your system
2 #include "Point2df.h"
3 #include "utility.h"
4 #include <math.h>
5 #include <GL/freeglut.h>
6 #include <math.h>
7 #include <float.h>
8 #include <vector>
9 #include <stdio.h>
10 using namespace std;
11 vector<Point2df>m_points;
12 typedef struct Rect2dfs
13 {
14     Point2df left;
15     float width;
16     float height;
```



```

57 // tu certili aris pirvel martkutxedshi
58 // daxatavt certils citlad
59 // tu aris meore martkutxedshi
60 //daxatavt lurjad
61 // magaliti
62 glColor3f(1.0f, 0.0f, 0.0f);
63 glVertex2f(m_points[0].x,m_points[0].y);
64 glEnd();
65 glFlush();
66 }
67 //<<<<<<<<<< main >>>>>>>>>
68 int main(int argc, char** argv)
69 {
70     glutInit(&argc, argv);
71     // initialize the toolkit
72     glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
73     // set display mode
74     glutInitWindowSize(screenWidth, screenHeight);
75     // set window size
76     glutInitWindowPosition(100, 150);
77     // set window position on screen
78     glutCreateWindow("Point in Rectangle");
79     // aq m1 da m2 martkutxedebis zomebi (sackisi certili
80     // sigane simagle) miecit
81     // magaliti
82     m1.left.set(1,1);
83     m1.width=40;
84     m1.height=20;
85
86     //aq gamoidzakhet funqcia romelic agenerirebs
87     ↪ shemtxvevit
88     // certilebs.
89 //magalitud
89 m_points.push_back(Point2df(-10.5,30.5));
90
91 // open the screen window
92 glutDisplayFunc(myDisplay);
93 // register redraw function
94 myInit();
95 glutMainLoop();

```

```

96 // go into a perpetual loop
97 return 1;
98 }

```

თქვენი ამოცანაა:

1. შექმენით ახალი პროექტი.
2. შექმენით შემთხვევითად გენერირებული 7 წერტილი, რომელიც შენახული იქნება ვექტორ მასივში `m_points`.
3. შექმენით ორი მართკუთხედი.
4. დახატეთ შავ ფერში.
5. ყოველი წერტილისთვის შეამოწმეთ: თუ წერტილი არის პირველ მართკუთხედში დახატეთ წერტილი წითელ ფერში. თუ მეორეში — ლურჯ ფერში. თუ არცერთ მართკუთხედში — მწვანე ფერში

2 წერტილი მართკუთხედში

5 ქულა

წინა ამოცანაში მართკუთხედი მოცემულია სტრუქტურის სახით.

1. შექმენით ახალი პროექტი და დაამატეთ საჭირო ფაილები/კლასები გამოყენებული პროექტებიდან.
 2. შექმენით მართკუთხედის კლასი.
 3. მართკუთხედის კლასში დაამატეთ ფუნქცია `bool testPoint(class Point2Df& p)`, რომელიც შეამოწმებს არის თუ არა წერტილი მართკუთხედის შიგნით.
 4. გააკეთეთ პირველი ამოცანა მართკუთხედის კლასის გამოყენებით.
- ყველა პროექტი უნდა მუშაობდეს. ანუ უნდა ეშვებოდეს პროგრამა და ხატავდეს.