



# מבחן בקורס תכנות ++C לתלמידי כימיה - 76635

מועד א' סמסטר א' תשס"ג

כל חומר כתוב מותר בשימוש  
משך הזמן שעתיים

הנה חלק מההגדרה של מחלקה בשם Point המייצגת מספר נקודה במישור. במחלקה 2 משתנים מסוג double בשם X ו Y המציינים את מיקום הנקודה יחסית לראשית הצירים. כמו כן מכילה המחלקה שני מצביעים ל double ומשתנים מסוג int שימשו להחזקת ההיסטוריה של מיקום הנקודה (כמערכים שגודלם נקבע דינמית)

```
01 class point
02 {
03 private:
04     double x, y
05     int histsize, histi;
06     double *xhist, *yhist;
07 };
```

1. הגדירו constructor למחלקה. הפונקציה תקבל 3 פרמטרים: מיקום הנקודה, וגודל מערכי ההיסטוריה הנדרשים.
2. הגדירו constructor ברירת מחדל למחלקה. הפונקציה תאפס את כל המשתנים הרלוונטיים ותגדיר היסטוריה בגודל 5.
3. הגדירו את האופרטור + לחיבור בין שתי נקודות. נקודת התוצאה תכיל מערכי היסטוריה ריקים בגודל מערכי ההיסטוריה הגדולים מבין המחברים.
4. הגדירו פונקציה של המחלקה movexy שתקבל 2 משתנים x ו y ותעביר את הנקודה למיקום xy לאחר שמירת המיקום הנוכחי בהיסטוריה. אם מערכי ההיסטוריה מלאים ימחק המיקום הישן ביותר, כל האחרים יזונו ויפנו מקום.
5. הגדירו פונקציה למחלקה המחזירה את מרחק הנקודה מראשית הצירים.
6. הגדירו פונקציה למחלקה בשם showhist המדפיסה את ההיסטוריה של נקודה למסך.
7. הגדירו אופרטור השמה למחלקה.
8. הגדירו copy constructor למחלקה.
9. הגדירו destructor למחלקה.
10. העמיסו את אופרטור ההחזרה >> לקלט מלוח המקשים עבור המחלקה.
11. העמיסו את אופרטור הוצאה << לפלט למסך עבור המחלקה (כול האינפורמציה כולל ההיסטוריה)

לא צריך main ☺.

בהצלחה

```

01 /*****
02 File: E:\DOC\CPP2003\test1\main.cpp
03 Revision By: Benjamin Czaczkes PhD
04 Revised on 19/01/2003 15:49:05
05 *****/
06
07 #include <iostream>
08 #include <math.h>
09 using namespace std;
10
11 class point
12 {
13 private:
14     double x, y;           // point x , y
15     int histsize,         // point history size
16     histi;                // number of history entries
17     double *xhist, *yhist; // point x,y history
18 public:
19     point(double X, double Y, int Histsize); //q1
20     point(void); //q2
21     const point operator+(const point & rhs); //q3
22     void movexy(double X, double Y); //q4
23     double size(void) //q5
24     {return sqrt(x*x+y*y);} //q5
25     void showhist(void) const; //q6
26     point & operator=(const point & rhs); //q7
27     point(const point & rhs); //q8
28     ~point(void) //q9
29     {delete [] xhist; delete [] yhist;} //q9
30     friend istream & operator>> //q10
31     (istream & lhs, point & rhs); //q10
32     friend ostream & operator<< //q11
33     (ostream & lhs, const point & rhs); //q11
34 };
35 // not part of the answer just to see that everything works
36 int main()
37 {
38     point p;
39     point pp(5,5,10);
40     cout << p << endl;
41     cout << pp << endl;
42     p=pp;
43     cout << p << endl;
44     p.movexy(2,1); cout << p << endl;
45     p.movexy(3,2); cout << p << endl;
46     point ppp=p;
47     cout << ppp << endl;
48     p.movexy(4,3); cout << p << endl;
49     p.movexy(5,4); cout << p << endl;
50     p.movexy(6,5); cout << p << endl;
51     p.movexy(7,6); cout << p << endl;
52     p=pp+p; cout << "p+pp 7,6+5,5=" << p << endl;
53     cout << "x y s i [xh[i] xy[i]]" << endl;
54     cin >> p;
55     cout << p;
56     return 0;
57 }
58 // end of not part of the answer
59 point::point(double X, double Y, int Histsize) //q1
60 {
61     histsize=Histsize; histi=-1;
62     x=X; y=Y;
63     xhist = new double [histsize];
64     yhist = new double [histsize];
65 }
66
67 point::point(void) //q2
68 {
69     histsize=5; histi=-1;
70     x=0; y=0;
71     xhist = new double [histsize];
72     yhist = new double [histsize];
73 }
74
75 const point point::operator+(const point & rhs) //q3
76 {
77     int Histsize=histsize;
78     if (histsize<rhs.histsize) Histsize=rhs.histsize;
79     return point(x+rhs.x, y+rhs.y, Histsize);
80 }

```

```

"E:\DOC\CPP2003\test1\Debug\test1.exe"
x=0 y=0 s=5 i=-1
x=5 y=5 s=10 i=-1
x=5 y=5 s=10 i=-1
x=2 y=1 s=10 i=0
i=0 x=5 y=5
i=1 x=2 y=1
x=3 y=2 s=10 i=1
i=0 x=5 y=5
i=1 x=2 y=1
x=4 y=3 s=10 i=2
i=0 x=5 y=5
i=1 x=2 y=1
i=2 x=3 y=2
x=5 y=4 s=10 i=3
i=0 x=5 y=5
i=1 x=2 y=1
i=2 x=3 y=2
i=3 x=4 y=3
x=6 y=5 s=10 i=4
i=0 x=5 y=5
i=1 x=2 y=1
i=2 x=3 y=2
i=3 x=4 y=3
i=4 x=5 y=4
x=7 y=6 s=10 i=5
i=0 x=5 y=5
i=1 x=2 y=1
i=2 x=3 y=2
i=3 x=4 y=3
i=4 x=5 y=4
i=5 x=6 y=5
p+pp 7,6+5,5= x=12 y=11 s=10 i=-1
x y s i [xh[i] xy[i]]
89 98 5 2 1 2 3 4 5 6 99 99 99 99
x=89 y=98 s=5 i=2
i=0 x=1 y=2
i=1 x=3 y=4
i=2 x=5 y=6
Press any key to continue_

```

```

81
82 void point::movexy(double X, double Y) //q4
83 {
84     if(histi<histsize-1)
85     {
86         histi++;
87     }
88     else
89     {
90         for(int i=0;i<histi;i++)
91         {
92             xhist[i]=xhist[i+1];
93             yhist[i]=yhist[i+1];
94         }
95     }
96     xhist[histi]=x;
97     yhist[histi]=y;
98     x=X; y=Y;
99 }
100
101 void point::showhist(void) const //q6
102 {
103     for(int i=0; i<=histi; i++)
104         cout<< " i=" << i
105             << " x=" << xhist[i]
106             << " y=" << yhist[i]
107             << endl;
108 }
109
110 point & point::operator=(const point & rhs) //q7
111 {
112     if (this!=&rhs)
113     {
114         delete [] xhist; delete [] yhist;
115         histsize=rhs.histsize;
116         histi=rhs.histi;
117         xhist = new double [histsize];
118         yhist = new double [histsize];
119         for(int i=0;i<=histi;i++)
120         {
121             xhist[i]=rhs.xhist[i];
122             yhist[i]=rhs.yhist[i];
123         }
124         x=rhs.x;
125         y=rhs.y;
126     }
127     return *this;
128 }
129
130 point::point (const point & rhs) //q8
131 {
132     histsize=rhs.histsize;
133     histi=rhs.histi;
134     xhist = new double [histsize];
135     yhist = new double [histsize];
136     for(int i=0;i<=histi;i++)
137     {
138         xhist[i]=rhs.xhist[i];
139         yhist[i]=rhs.yhist[i];
140     }
141     x=rhs.x;
142     y=rhs.y;
143     return;
144 }
145
146 istream & operator>>(istream & lhs, point & rhs) //q10
147 {
148     lhs >> rhs.x
149         >> rhs.y
150         >> rhs.histsize
151         >> rhs.histi;
152     delete [] rhs.xhist;
153     delete [] rhs.yhist;
154     rhs.xhist = new double [rhs.histsize];
155     rhs.yhist = new double [rhs.histsize];
156     for(int i=0;i<=rhs.histi;i++)
157         lhs >> rhs.xhist[i] >> rhs.yhist[i];
158     return lhs;
159 }
160
161 ostream & operator<<(ostream & lhs, const point & rhs) //q11
162 {
163     lhs << " x=" << rhs.x
164         << " y=" << rhs.y
165         << " s=" << rhs.histsize
166         << " i=" << rhs.histi
167         << endl;
168     rhs.showhist();
169     return lhs;
170 }

```