

Linux day2 hw2 결과 보고서

2025407012/로봇학부/송연우

목차

1. hpp파일
2. cpp파일
3. 실행 결과

1. hpp 파일

다음은 작성한 헤더파일 코드입니다.

```
1 namespace SYU{
2     class GetDist{
3     public:
4         struct coordinate
5         {
6             int x;
7             int y;
8         };
9
10    private:
11        coordinate* dot;
12        int count;
13        int min_value;
14        int max_value;
15        double MinDist;
16        double MaxDist;
17        double Dist;
18        double p;
19        int check;
20        coordinate min1, min2, max1, max2;
21    public:
22        GetDist();
23        ~GetDist();
24        int GetCoord();
25        void PrintResult();
26        bool CinSecure(double p);
27
28    };
29
30 }
```

namespace SYU 안에 클래스를 선언 코드를 작성하였습니다. 구조체를 사용하기 위해 public에서 coordinate를 정의하고 private에서 변수를, public에서 함수를 선언했습니다. 구조체에는 int형 변수 x, y로 2차원 평면 좌표를 나타냈습니다. coordinate형으로 선언된 변수 중 *dot은 동적 메모리

할당을 위한 것이고, min1~max2는 최대거리/최소거리 값에서의 좌표들을 저장하기 위해 사용됩니다.

double형 MinDist, MaxDist, Dist, p는 제곱근의 소수점까지 나타내기 위해서와 정수형 입력에서 실수를 걸러내기 위해 선언되었습니다. 나머지 int형 변수들은 초반 사용자에게 입력받은 값을 검사 후 저장하는 용도로 선언했습니다. check의 경우에는 p와 함께 정수형 검사에 사용했습니다.

다음으로 함수들입니다. GetDist(), ~GetDist()함수는 각각 생성자, 소멸자로 각각 기본값을 지정하고 메모리를 해제합니다. GetCoord()에서는 사용자로부터 생성할 점의 개수, 좌표 최솟값, 좌표 최댓값을 입력받고, 잘못된 값을 입력하면 오류 메시지를 띄웁니다. 그리고 rand()함수로 사용자가 입력한 만큼의 점을 최솟값과 최댓값 사이에서 생성하고, 동적 메모리 할당을 받은 dot에 저장합니다.

PrintResult()에서는 dot에 저장된 값들을 계산해 최소 거리와 최대 거리를 구하고, 결과를 출력합니다. CinSecure(double p)에서는 사용자가 입력한 실수형 변수에 저장된 값을 매개변수로 받고 정수형으로 변환 후 다시 원 실수와 비교해 값이 다르면 false, 같으면 true를 반환합니다.

2. cpp 파일

다음은 작성한 소스코드입니다.

```
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <cmath>
#include <iomanip>
#include "hw2.hpp"

using namespace std;

namespace SYU{

    GetDist::GetDist(): dot(nullptr), count(0), min_value(0), max_value(0), MinDist(0), MaxDist(0), Dist(0) {}
    GetDist::~GetDist() {delete[] dot;}

    bool GetDist::CinSecure(double p){
        check = p;
        if(check!=p){
            return false;
        }
        return true;
    }
}
```

GetDist()함수에서는 기본값을 정해주고 ~ GetDist()함수에서는 메모리를 해제합니다.

CinSecure(double p)는 실수형과 실수형을 복사한 정수형 변수의 값을 비교해 실수 입력을 걸러내는 역할을 합니다.

```

22     int GetDist::GetCoord(){
23         cout<<"Please define the number of points: ";
24         cin>>p;
25         if(!CinSecure(p)){
26             cout<<"정수만을 입력할 수 있습니다."<<endl;
27             return 0;
28         }
29         count=p;
30
31         cout<<"Please define minimum of coor. value: ";
32         cin>>p;
33         if(!CinSecure(p)){
34             cout<<"정수만을 입력할 수 있습니다."<<endl;
35             return 0;
36         }
37         min_value=p;
38
39         cout<<"Please define maximum of coor. value: ";
40         cin>>p;
41         if(!CinSecure(p)){
42             cout<<"정수만을 입력할 수 있습니다."<<endl;
43             return 0;
44         }
45         max_value=p;
46
47         if(count<=0||(min_value>=max_value)){
48             cout<<"잘못된 값을 입력했습니다."<<endl;
49             return 0;
50         }
51     }

```

GetCoord()함수입니다. 생성할 점의 개수, 최소값, 최대값을 입력받는 부분입니다. 사용자가 입력한 값이 실수면 오류 메시지를 출력하고 정수면 각각의 정수형 변수에 p값을 저장한 뒤 마지막에 count가 0이하이거나 최솟값이 최댓값보다 크거나 같은 경우 오류 메시지를 출력합니다. 오류 메시지를 출력한 뒤에는 모두 0을 반환해 main문에서 활용합니다.

```

52     srand(time(NULL));
53     dot=new coordinate[count];
54
55     for(int i=0;i<count;i++){
56
57         int num1 = rand()%max_value;
58         int num2 = rand()%max_value;
59
60         if(num1<min_value||num2<min_value){
61             i--;
62             continue;
63         }
64         dot[i].x = num1;
65         dot[i].y = num2;
66         cout<<"Point "<<i+1<<'. '<< "nX="<<num1<<','<<"nY="<<num2<<endl;
67     }
68 }
69

```

GetCoord()함수의 뒷부분입니다. 랜덤으로 좌표를 생성할 수 있도록 시드를 초기화하고 동적 메모리를 할당합니다. 랜덤으로 좌표를 생성할 때는 랜덤값을 max_value로 나눈 나머지 값을 사용하고, 나머지가 min_value보다 크거나 같은 경우만을 사용합니다. 한 번 값을 뽑을 때마다 dot에 저장하고 출력합니다.

```
void GetDist::PrintResult(){
    MinDist=((double)dot[0].x-(double)dot[1].x)*((double)dot[0].x-(double)dot[1].x)+((double)dot[0].y-(double)dot[1].y)*((double)dot[0].y-(double)dot[1].y);
    MinDist=sqrt(MinDist);
    MaxDist=MinDist;
    min1=dot[0];min2=dot[1];
    max1=min1;max2=min2;
    for (int i=0;i<count;i++){
        for(int j=0;j<count;j++){
            if(i==j)continue;
            Dist=((double)dot[i].x-(double)dot[j].x)*((double)dot[i].x-(double)dot[j].x)+((double)dot[i].y-(double)dot[j].y)*((double)dot[i].y-(double)dot[j].y);
            Dist = sqrt(Dist);
            if(Dist<MinDist){MinDist=Dist;min1=dot[i];min2=dot[j];}
            if(Dist>MaxDist){MaxDist=Dist;max1=dot[i];max2=dot[j];}
        }
    }

    cout << fixed << setprecision(6);

    cout<<"----- Result -----"<<endl;
    cout<<"MinDist="<<MinDist<<endl;
    cout<<"Pair of Min Coor.(x,y): P1("<<min1.x<<','<<min1.y<<") & P2("<<min2.x<<','<<min2.y<<')<<endl;

    cout<<"MaxDist="<<MaxDist<<endl;
    cout<<"Pair of Max Coor.(x,y): P1("<<max1.x<<','<<max1.y<<") & P2("<<max2.x<<','<<max2.y<<')<<endl;

    cout<<"***** Completed *****"<<endl;

}
}
```

PrintResult()함수입니다. MinDist, MaxDist는 우선 0번 점과 1번 점 사이의 거리로 하고 min1~max2값도 임시로 정해줍니다. 다음 간단한 선택 정렬로 Dist를 계산하고 최소거리와 최대거리를 갱신합니다. 다음으로 결과값을 출력할 때 소수점까지 출력하기 위해서 cout<<fixed<<setprecision(6) 코드를 추가해 소수점 아래 6번째까지 출력해 줍니다.

3. 실행 결과

```
yu@yu:~/intern_ws/hw2/build$ ./test
Please define the number of points: 4
Please define minimum of coor. value: -1
Please define maximum of coor. value: 10
Point 1.nX=0,nY=7
Point 2.nX=3,nY=7
Point 3.nX=0,nY=1
Point 4.nX=1,nY=6
----- Result -----
MinDist=1.414214
Pair of Min Coor.(x,y): P1(0,7) & P2(1,6)
MaxDist=9.000000
Pair of Max Coor.(x,y): P1(0,7) & P2(3,7)
***** Completed *****
yu@yu:~/intern_ws/hw2/build$ ./test
```

위는 정상 작동할 때의 실행 결과입니다.

```
yu@yu:~/intern_ws/hw2/build$ ./test
Please define the number of points: 3.4
정수만을 입력할 수 있습니다.
Please define the number of points: -1
Please define minimum of coor. value: 3
Please define maximum of coor. value: 7
잘못된 값을 입력했습니다.
Please define the number of points: 4
Please define minimum of coor. value: 10
Please define maximum of coor. value: -1
잘못된 값을 입력했습니다.
Please define the number of points: -1
Please define minimum of coor. value: 10
Please define maximum of coor. value: -1
잘못된 값을 입력했습니다.
Please define the number of points: 4
Please define minimum of coor. value: -10
Please define maximum of coor. value: 10
Point 1.nX=6,nY=1
Point 2.nX=0,nY=6
Point 3.nX=9,nY=0
Point 4.nX=4,nY=6
----- Result -----
MinDist=3.162278
Pair of Min Coor.(x,y): P1(6,1) & P2(9,0)
MaxDist=61.000000
Pair of Max Coor.(x,y): P1(6,1) & P2(0,6)
***** Completed *****
yu@yu:~/intern_ws/hw2/build$
```

예외처리 부분입니다. 음수를 입력하거나 실수를 입력하면 오류 메시지를 출력합니다.