```cpp
#include"dm.h"

/// member functions of circle class

void circle::get_information()
{

     cout<<"NAME: "<<setw(8)<<name<<", "<<"PERIMETER: "
     <<setw(8)<<perimeter<<", "<<"TYPE: circle"<<endl;
}
circle::circle(double r,string s1)
{
    r1=r;
    name=s1;
}
void circle::set_perimeter()
{
    if(r1<0)
    {
        perimeter=-1;
    }else{
        perimeter=2*3.14*r1;
    }

}

///member functions of triangle class
void triangle::get_information()
{
     cout<<"NAME: "<<setw(8)<<name<<", "<<"PERIMETER: "
     <<setw(8)<<perimeter<<", "<<"TYPE: triangle"<<endl;
}
triangle::triangle(double side0,double side1,double side2,string s1
{
    sides=new double[3];
    sides[0]=side0;
    sides[1]=side1;
    sides[2]=side2;
    name=s1;

}
void triangle::set_perimeter()
{
    if(sides[0]+sides[1]<=sides[2]||sides[0]+sides[2]<=sides[1]||si
   {
        perimeter=-1;
```

```
47:     }
48:     else{
49:         perimeter=sides[0]+sides[1]+sides[2];
50:     }
51: }
52:
53:
54: ///member functions of rectangle class
55: void rectangle::get_information()
56: {
57:
58:      cout<<"NAME: "<<setw(8)<<name<<", "<<"PERIMETER: "
59:      <<setw(8)<<perimeter<<", "<<"TYPE: rectangle"<<endl;
60: }
61: rectangle::rectangle(double side0,double side1,double side2,double s
62: {
63:     sides=new double[4];
64:     sides[0]=side0;
65:     sides[1]=side1;
66:     sides[2]=side2;
67:     sides[3]=side3;
68:     name=s1;
69:
70: }
71: void rectangle::set_perimeter()
72: {
73:     if(sides[0]==sides[2]&&sides[1]==sides[3])
74:    {
75:         perimeter=sides[0]+sides[1]+sides[2]+sides[3];
76:     }
77:     else{
78:         perimeter=-1;
79:     }
80: }
81:
82: /// member functions of square class
83: void square::get_information()
84: {
85:      cout<<"NAME: "<<setw(8)<<name<<", "<<"PERIMETER: "
86:      <<setw(8)<<perimeter<<", "<<"TYPE: square"<<endl;
87: }
88:
89: square::square(double side0,double side1,double side2,double side3,s
90: {
91:     sides=new double[4];
92:     sides[0]=side0;
```

```
 93:     sides[1]=side1;
 94:     sides[2]=side2;
 95:     sides[3]=side3;
 96:     name=s1;
 97: }
 98: void square::set_perimeter()
 99: {
100:     if(sides[0]==sides[1]&&sides[1]==sides[2]&&sides[2]==sides[3])
101:    {
102:         perimeter=sides[0]+sides[1]+sides[2]+sides[3];
103:     }
104:     else{
105:         perimeter=-1;
106:     }
107: }
108:
109:
110:
111:
112:
```