```
 1: #ifndef DM_H
 2: #define DM_H
 3: #include <iostream>
 4: #include <fstream>
 5: #include <string>
 6: #include <iomanip>
 7: #define pi 3.14
 8: using namespace std;
 9:
10: ///shape class
11: class shape
12: {
13:       public:
14:             virtual ~shape(){};
15:             virtual void get_information()=0;
16:             virtual void set_perimeter()=0;
17:             double get_perimeter(){return perimeter;}
18:       protected:
19:             double perimeter;
20:             string name;
21:
22: };
23:
24: ///polygon class
25: class polygon:public shape
26: {
27:       public:
28:              ~polygon(){delete[] sides;}
29:       protected:
30:             double* sides;
31:
32: };
33:
34: ///circle class
35: class circle:public shape
36: {
37:     public:
38:         void get_information();
39:         void set_perimeter();
40:         circle(double ,string );
41:     private:
42:         double r1;
43:
44: };
45:
46:
```

```
47: ///triangle class
48: class triangle:public polygon
49: {
50:     public:
51:         void get_information();
52:         void set_perimeter();
53:         triangle(double,double,double,string);
54: };
55:
56: ///rectangle class
57: class rectangle:public polygon
58: {
59:     public:
60:         void get_information();
61:         void set_perimeter();
62:         rectangle(double,double,double,double,string);
63:
64:
65:
66: };
67:
68: ///square class
69: class square:public polygon
70: {
71:     public:
72:         void get_information();
73:         void set_perimeter();
74:         square(double,double,double,double,string);
75:
76:     //square():rectangle(){}
77:     //square(double s_0,double s_1,double s_2,double s_3,string n)
78:     //:rectangle(s_0, s_1, s_2, s_3, n){};
79:     //void get_information();
80:     //void set_perimeter();
81:
82: };
83: #endif
84:
85:
```