```cpp
#include <iostream>
#include "Shapes.h"


int main() {
    // Trying to declare this will give an error
    // because it's not allowed to instantiate an
    // object of an abstract class.

    Rectangle rect(3 , 4);
    Square q(3);
    Circle c(3);


    std::cout << "Rectangle: [" << rect.height() << " , " << rect.width() << "]"
              << std::endl;
    std::cout << "Square : ["   <<   q.height()  << " , " <<   q.width()  << "]"
              << std::endl;
    std::cout << "Circle : ["   <<   c.height()  << " , " <<   c.width()  << "]\n"
              << std::endl;


    std::cout << "Area: "         << '\n';
    std::cout << "+ Rectangle : " << rect.area()  << std::endl;
    std::cout << "+ Square : "    <<   q.area()   << std::endl;
    std::cout << "+ Circle : "    <<   c.area()   << '\n' << std::endl;


    std::cout << "Perimeter: "    << '\n';
    std::cout << "+ Rectangle : " << rect.perimeter() << std::endl;
    std::cout << "+ Square : "    <<  q.perimeter()   << std::endl;
    std::cout << "+ Circle : "    <<  c.perimeter()   << '\n' << std::endl;


    std::cout << "Height and width: " << '\n';
    std::cout << "+ Rectangle : " << rect.height() << " , " << rect.width()
              << std::endl;
    std::cout << "+ Square : "    <<   q.height()  << " , " <<  q.width()
              << std::endl;
    std::cout << "+ Circle : "    <<   c.height()  << " , " <<  c.width() << '\n'
              << std::endl;

    // Test rotation
    Rectangle r(2 , 7);
    std::cout << "Rectangle:" << std::endl;
    std::cout << "[" << r.height() << " , " << r.width() << "] rotation -> ";

    r.rotate();

    std::cout << "[" << r.height() << " , " << r.width() << "]" << std::endl;

    std::cout << "Square:" << std::endl;
    std::cout << "[" << q.height() << " , " << q.width() << "] rotation -> ";

    q.rotate();

    std::cout << "[" << q.height() << " , " << q.width() << "]" << std::endl;

```

```
59
60
61      std::cout << "Circle:" << std::endl;
62      std::cout << "[" << c.height() << " , " << c.width() << "] rotation -> ";
63
64      c.rotate();
65
66      std::cout << "[" << c.height() << " , " << c.width() << "]" << std::endl;
67
68      return 0;
69  }
```