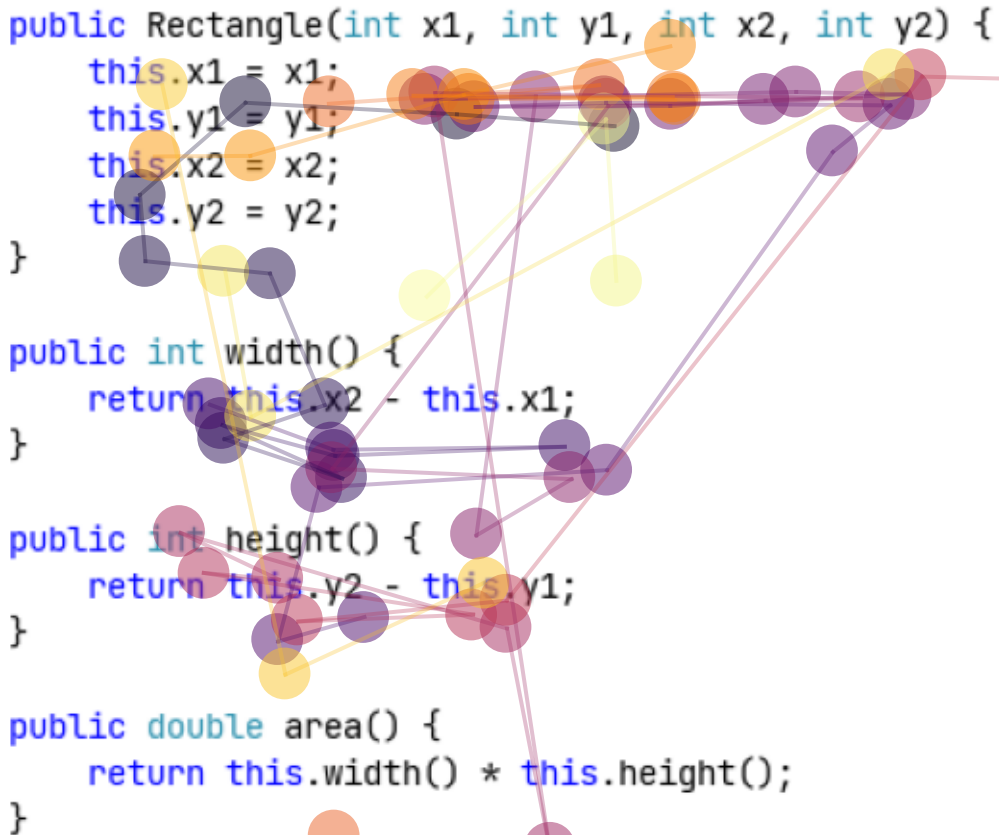


```
public Rectangle(int x1, int y1, int x2, int y2) {  
    this.x1 = x1;  
    this.y1 = y1;  
    this.x2 = x2;  
    this.y2 = y2;  
}  
  
public int width() {  
    return this.x2 - this.x1;  
}  
  
public int height() {  
    return this.y2 - this.y1;  
}  
  
public double area() {  
    return this.width() * this.height();  
}
```



The image displays the source code of a Java class named `Rectangle` with four methods: a constructor, `width()`, `height()`, and `area()`. Overlaid on the code is a complex graph representing the Abstract Syntax Tree (AST) and control flow graph. Nodes, represented by colored circles (yellow, orange, purple, red, and grey), are connected by lines indicating the flow of execution and the structure of the code. The graph shows how the code is parsed into a hierarchical tree structure and how the execution flow moves from the constructor to the methods and between the methods.