

信息科学技术学院课程作业

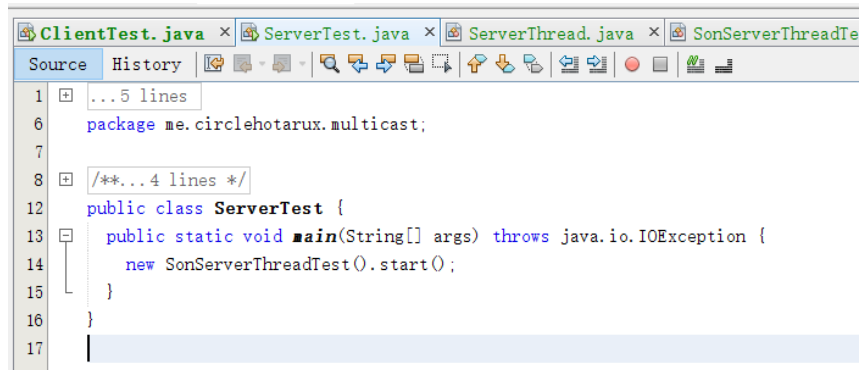
得分： 分

课程名称	网络通信程序设计	班级	17 软 2
作业内容	组播编程作业	任课教师	肖东
姓名	陈庆辉	学号	1714080902201
提交日期	2020/5/19		

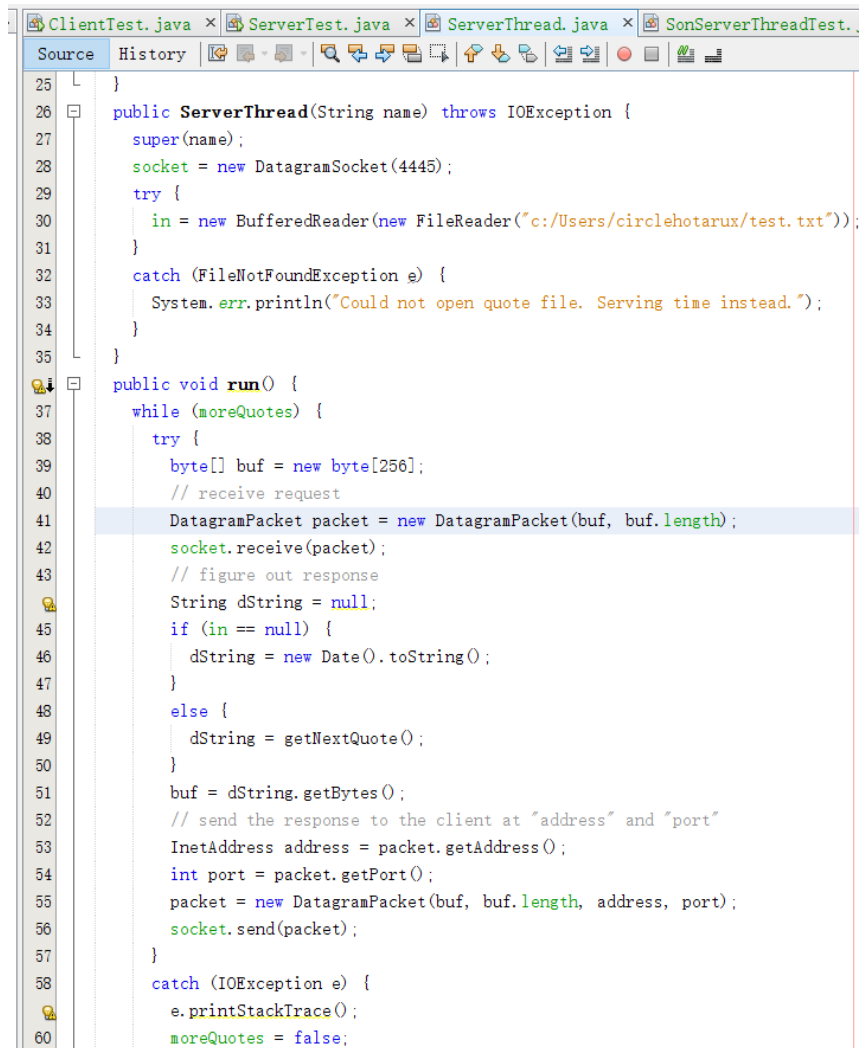
题目 1：面向 UDP 编程作业

Blog 地址：<https://blog.csdn.net/xdong001>

(1) 创建组播发送（服务）端，代码截图：



```
1 ... 5 lines
6 package me.circlehotarux.multicast;
7
8 /**... 4 lines */
12 public class ServerTest {
13     public static void main(String[] args) throws java.io.IOException {
14         new SonServerThreadTest().start();
15     }
16 }
17
```



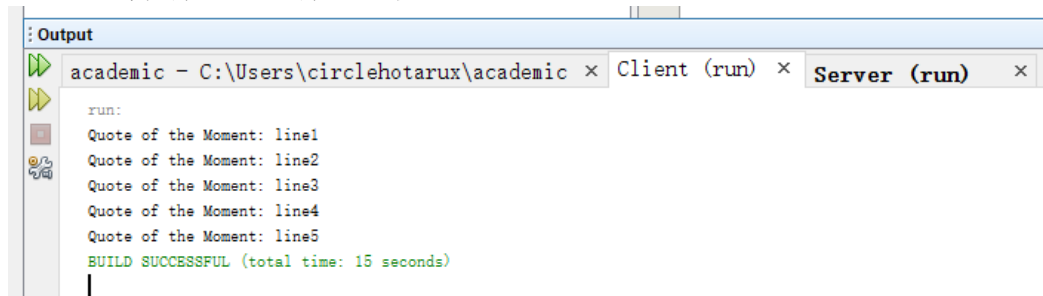
```
25 }
26 public ServerThread(String name) throws IOException {
27     super(name);
28     socket = new DatagramSocket(4445);
29     try {
30         in = new BufferedReader(new FileReader("c:/Users/circlehotarux/test.txt"));
31     }
32     catch (FileNotFoundException e) {
33         System.err.println("Could not open quote file. Serving time instead.");
34     }
35 }
36 public void run() {
37     while (moreQuotes) {
38         try {
39             byte[] buf = new byte[256];
40             // receive request
41             DatagramPacket packet = new DatagramPacket(buf, buf.length);
42             socket.receive(packet);
43             // figure out response
44             String dString = null;
45             if (in == null) {
46                 dString = new Date().toString();
47             }
48             else {
49                 dString = getNextQuote();
50             }
51             buf = dString.getBytes();
52             // send the response to the client at "address" and "port"
53             InetAddress address = packet.getAddress();
54             int port = packet.getPort();
55             packet = new DatagramPacket(buf, buf.length, address, port);
56             socket.send(packet);
57         }
58         catch (IOException e) {
59             e.printStackTrace();
60             moreQuotes = false;
61         }
62     }
63 }
```

```
ClientTest.java x ServerTest.java x ServerThread.java x SonServerThreadTest.java
Source History
16 class SonServerThreadTest
17 extends ServerThread {
18     private long FIVE_SECONDS = 5000;
19     public SonServerThreadTest() throws IOException {
20         super("SonServerThreadTest");
21     }
22
23     public void run() {
24         while (moreQuotes) {
25             try {
26                 byte[] buf = new byte[256];
27                 // construct quote
28                 String dString = null;
29                 if (in == null) {
30                     dString = new Date().toString();
31                 }
32                 else {
33                     dString = getNextQuote();
34                 }
35                 buf = dString.getBytes();
36                 // send it
37                 InetAddress group = InetAddress.getByName("224.0.2.8");
38                 DatagramPacket packet = new DatagramPacket(buf, buf.length, group,
39                     4446);
40                 socket.send(packet);
41                 // sleep for a while
42                 try {
43                     sleep((long) (Math.random() * FIVE_SECONDS));
44                 }
45                 catch (InterruptedException e) {}
46             }
47             catch (IOException e) {
48                 e.printStackTrace();
49                 moreQuotes = false;
50             }
51         }
52     }
53 }
```

(2) 创建组播接收（客户）端，代码截图：

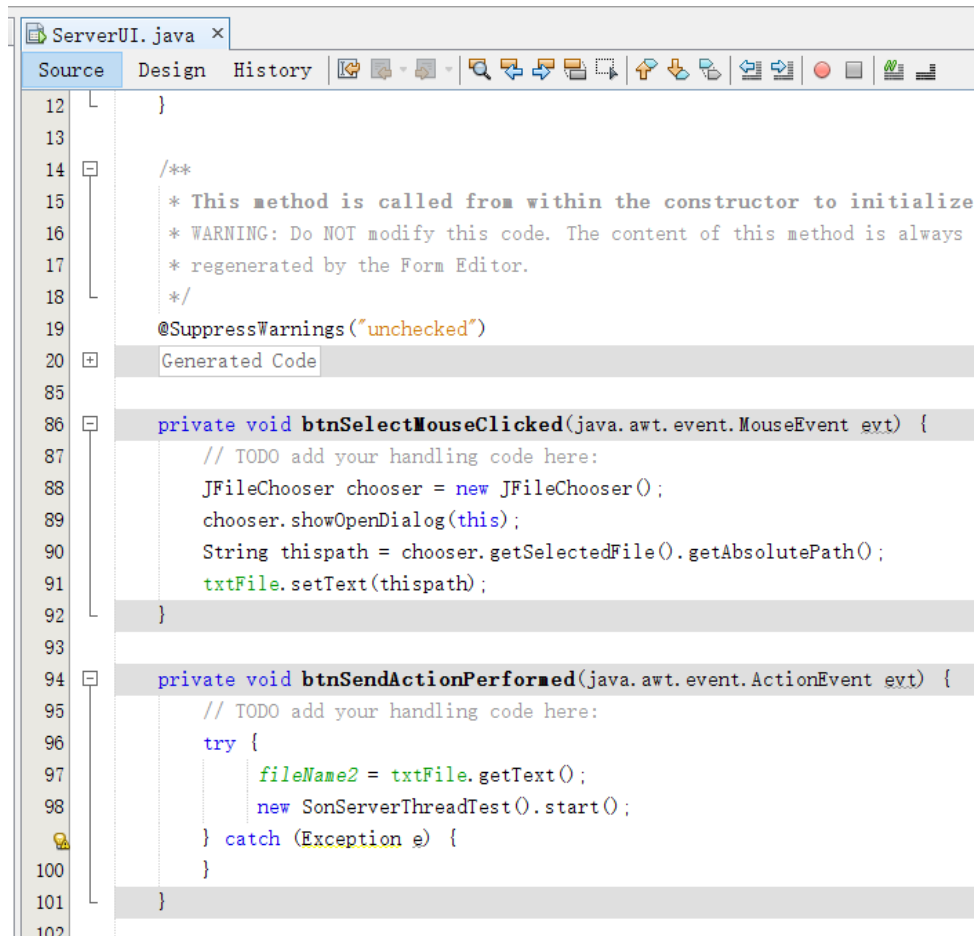
```
ClientTest.java x ServerTest.java x ServerThread.java x So
Source History
1 ... 5 lines
6 package me.circlehotarux.multicast;
7
8 import ... 2 lines
10 /**... 4 lines */
14
15 public class ClientTest {
16     public static void main(String[] args) throws IOException {
17         MulticastSocket socket = new MulticastSocket(4446);
18         InetAddress address = InetAddress.getByName("224.0.2.8");
19         socket.joinGroup(address);
20         DatagramPacket packet;
21         for (int i = 0; i < 5; i++) {
22             byte[] buf = new byte[256];
23             packet = new DatagramPacket(buf, buf.length);
24             socket.receive(packet);
25             String received = new String(packet.getData());
26             System.out.println("Quote of the Moment: " + received);
27         }
28         socket.leaveGroup(address);
29         socket.close();
30     }
31 }
```

(3) 服务端和客户端运行最终结果截图;



```
run:
Quote of the Moment: line1
Quote of the Moment: line2
Quote of the Moment: line3
Quote of the Moment: line4
Quote of the Moment: line5
BUILD SUCCESSFUL (total time: 15 seconds)
```

(4) 改良组播发送（服务）端程序，代码截图;



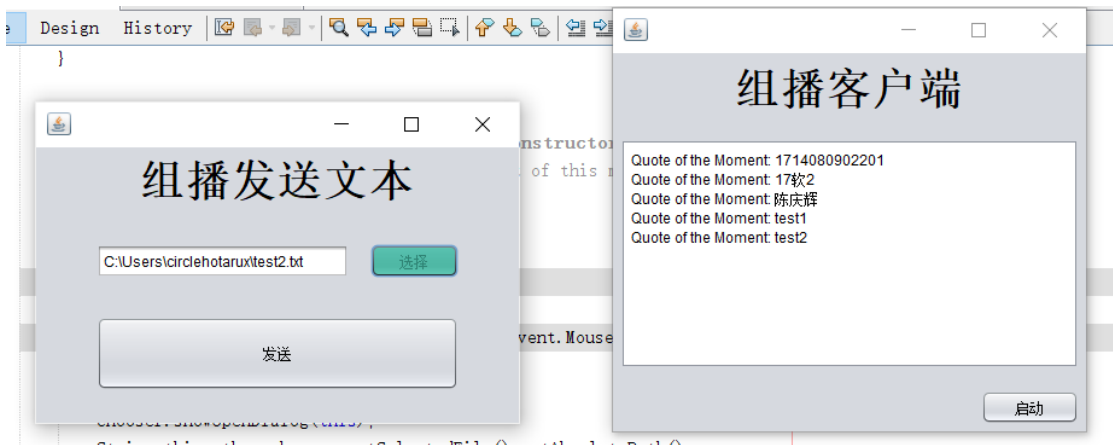
```
12 }
13
14 /**
15  * This method is called from within the constructor to initialize
16  * WARNING: Do NOT modify this code. The content of this method is always
17  * regenerated by the Form Editor.
18  */
19 @SuppressWarnings("unchecked")
20 Generated Code
85
86 private void btnSelectMouseClicked(java.awt.event.MouseEvent evt) {
87     // TODO add your handling code here:
88     JFileChooser chooser = new JFileChooser();
89     chooser.showOpenDialog(this);
90     String thispath = chooser.getSelectedFile().getAbsolutePath();
91     txtFile.setText(thispath);
92 }
93
94 private void btnSendActionPerformed(java.awt.event.ActionEvent evt) {
95     // TODO add your handling code here:
96     try {
97         fileName2 = txtFile.getText();
98         new SonServerThreadTest().start();
99     } catch (Exception e) {
100     }
101 }
102
```

```

24 }
25
26 /**
27  * This method is called from within the constructor to initialize the form
28  * WARNING: Do NOT modify this code. The content of this method is always
29  * regenerated by the Form Editor.
30  */
31 @SuppressWarnings("unchecked")
32 Generated Code
33
34
35
36 private void btnStartActionPerformed(java.awt.event.ActionEvent evt) {
37     // TODO add your handling code here:
38     try {
39         btnStart.setEnabled(false);
40         MulticastSocket socket = new MulticastSocket(4446);
41         InetAddress address = InetAddress.getByName("228.5.6.8");
42         socket.joinGroup(address);
43         DatagramPacket packet;
44         for (int i = 0; i < 5; i++) {
45             byte[] buf = new byte[256];
46             packet = new DatagramPacket(buf, buf.length);
47             socket.receive(packet);
48             String received = new String(packet.getData());
49             textArea.append("Quote of the Moment: " + received + "\n");
50         }
51         socket.leaveGroup(address);
52         socket.close();
53         btnStart.setEnabled(true);
54     } catch (IOException ex) {
55         textArea.append("不能打开数据报Socket，或数据报Socket无法与指定端口连接!");
56     }
57 }
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108

```

(5) 改良后最终结果截图：



题目 2：你是否请求同学帮助、指导同学操作，完成一个人实验。在实验中有没有组建团队？你的程序与其他同学的程序能连接上吗？

- ☐ A 我请求帮助的同学是：_____
- ☐ B 我帮助的同学是：_____
- ☐ C 我并没有帮助同学，也没有同学帮助我。
- ☐ D 我们团队名单是：_____1714080902221 郑润楷_____
- ☐ E 我的程序和其他同学的程序连接名单是：_____

考察内容	评分标准	满分	得分
具有建立软件模型、设计实验、分析与解释数据、并通过信息综合得到合理有效结论的能力。	完成题目 1 的所有操作，完成所有截图，每小题 10 分；	60	
具有软件工程项目管理、有效沟通、领域整合与团队合作的能力。	题目 2 中 A 得 10 分，B 得 5 分，C 得 0 分，D 得 5 分，E 得 10 分。	20	
具有设计针对复杂软件工程问题的解决方案，设计满足特定需求的软件系统的能力。	针对题目 1 中（4）的问题解决，其中程序代码的出错情况反映出这方面的能力。	20	
		合计	
简要点评：			