Heaven's light is our guide

Rajshahi University of Engineering & Technology, Bangladesh

Course No: CSE 2100

Course Title: Software Development Project I

Submitted to:	Submitted by:
Md. Ali Hossain Assistant Professor Dept. of CSE RUET	Md. Zobair Hussain Roll No: 143014 Dept. of CSE RUET

Offline Judgment

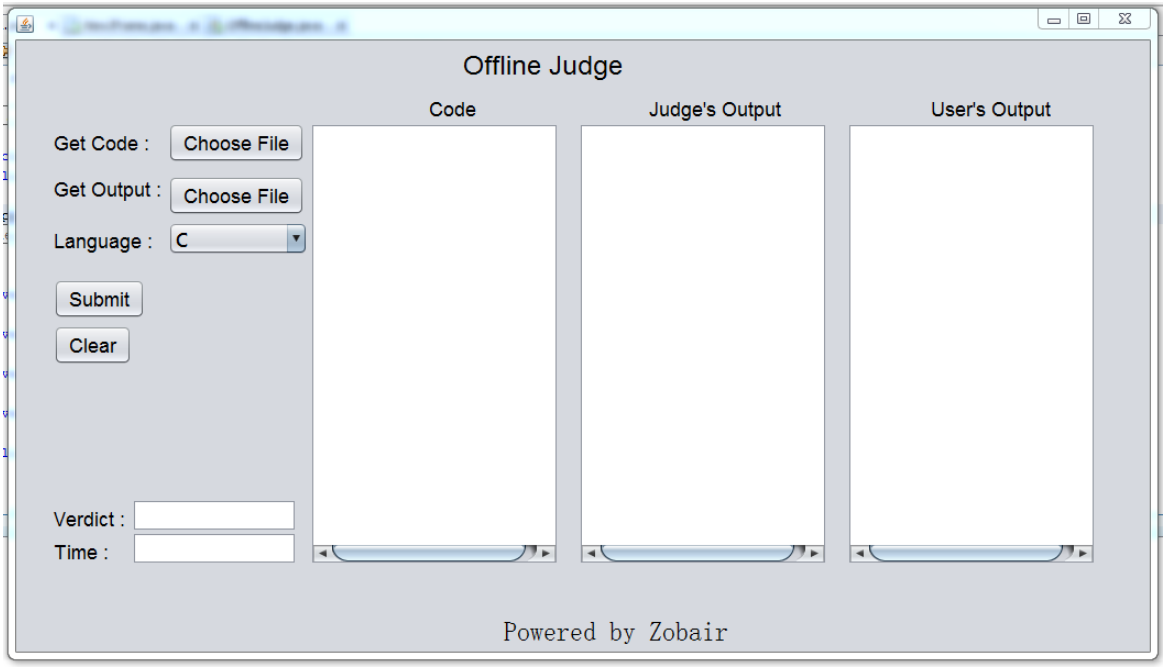
Purpose of the project: nowadays competitive programing is the biggest part of CSE subject. But all the judge system are online based. There are a few offline judge available in the market like PC^2. But it can't run in a single PC. It is vastly used to take onsite contest. So, there is no offline judge for single users. Here is the concept of my project. It is very simple for a single user to judge there code whether it is correct or not.

Required Apparatus: Net beans or eclipse or any kind of java runnable software.

Theory: this is a java based project. There is a editor. When we submit a code in the project, it edit the code in a while so that the code can create a txt file of output. Then we submit the original output ion the program. After that it checks the judge output & user's output line by line whether they are same or not. If they are same it gives accepted verdict & if not, then wrong answer.
Besides, it counts the total time to run the code. So that anyone can easily know that how much time his code took to run.

Project Description: (how it works)

01. There is a java code that takes the judge input & judge output form the user.



02. After that it convert the C or C++ or java code into a txt file.
03. There is another C++ code called output generator that edited that txt file in a way that, it can generate the output txt file.

output_generator.cpp | evaluator.cpp

```cpp
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int fl = 0;
    string str, s;
    ifstream in;
    ofstream out;
    in.open("submitted.txt");
    out.open("code.cpp");
    while(!in.eof())
    {
        getline(in, str);

        if (fl != 2)
        {
            stringstream strm;

            strm << str;

            while(strm >> s)
            {
                out << s << " ";
                if (s == "main" || s == "main(" ||
                {
                    fl = 1;
                    continue;
                }

                if (fl == 1)
                {
                    if (s == "{")
                    {
                        out << "freopen(\"input.txt
                        fl = 2;
```

04. Another code called evaluator check the judge output & user's output line by line. If they are same, the verdict is **Accepted**, if not, the verdict is **Wrong Answer**.

evaluator.cpp | output_generator.cpp

```cpp
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int t =6;
    FILE *in,*jg, *ou;

    in = fopen("output.txt","r");

    jg=fopen("judge.txt", "r");

    ou = fopen("verdict.txt", "w");

    char si[1000000], sj[1000000];

    while(t--)
    {
        fscanf(jg, "%s", sj);
//          getchar();
        fscanf(in, "%s", si);
  //        getchar();

    //      printf("t = %d\njudge = %s\ninput =

        if (strcmp(si, sj) != 0)
        {
            fprintf(ou, "Wrong Answer\n");
            fclose(in);
            fclose(jg);
            fclose(ou);
            return 0;
        }
    }
            fprintf(ou, "Accepted\n");
```

05. Besides it count the time taken by the code.
06. At the end, it shows the verdict and the time.

Code:

Main Code:

```
package p1;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Scanner;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JFileChooser;
import javax.swing.JFrame;

public class P1 extends javax.swing.JFrame {
   public P1() {
      initComponents();
   }

   @SuppressWarnings("unchecked")
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jScrollPane1 = new javax.swing.JScrollPane();
    jTextArea1 = new javax.swing.JTextArea();
    jScrollPane2 = new javax.swing.JScrollPane();
    jTextArea2 = new javax.swing.JTextArea();
    jScrollPane3 = new javax.swing.JScrollPane();
    jTextArea3 = new javax.swing.JTextArea();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jLabel6 = new javax.swing.JLabel();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jLabel9 = new javax.swing.JLabel();
    jLabel10 = new javax.swing.JLabel();
    jScrollPane4 = new javax.swing.JScrollPane();
    jTextPane1 = new javax.swing.JTextPane();
    jScrollPane5 = new javax.swing.JScrollPane();
    jTextPane2 = new javax.swing.JTextPane();
    jLabel11 = new javax.swing.JLabel();
    jButton3 = new javax.swing.JButton();
    jButton4 = new javax.swing.JButton();
    jLabel7 = new javax.swing.JLabel();
    jComboBox1 = new javax.swing.JComboBox();
    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setLocation(new java.awt.Point(150, 80));
    jLabel1.setFont(new java.awt.Font("Segoe UI Emoji", 0, 24)); // NOI18N
    jLabel1.setText("Offline Judge");
    jLabel2.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
    jLabel2.setText("Code");
    jTextArea1.setColumns(20);
    jTextArea1.setRows(5);
    jScrollPane1.setViewportView(jTextArea1);
    jTextArea2.setColumns(20);
    jTextArea2.setRows(5);
    jScrollPane2.setViewportView(jTextArea2);
    jTextArea3.setColumns(20);
```

```
        jTextArea3.setRows(5);
        jScrollPane3.setViewportView(jTextArea3);
        jLabel3.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jLabel3.setText("Judge's Output");
        jLabel4.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jLabel4.setText("User's Output");
        jLabel5.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jLabel5.setText("Get Code :");
        jLabel6.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jLabel6.setText("Get Output :");
        jButton1.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jButton1.setText("Choose File");
        jButton1.addActionListener(this::jButton1ActionPerformed);
        jButton2.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jButton2.setText("Choose File");
        jButton2.addActionListener(this::jButton2ActionPerformed);
        jLabel9.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jLabel9.setText("Verdict :");
        jLabel10.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jLabel10.setText("Time :");
        jScrollPane4.setViewportView(jTextPane1);
        jScrollPane5.setViewportView(jTextPane2);
        jLabel11.setFont(new java.awt.Font("SimSun-ExtB", 0, 24)); // NOI18N
        jLabel11.setText("Powered by Zobair");
        jButton3.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jButton3.setText("Clear");
        jButton3.addActionListener(this::jButton3ActionPerformed);
        jButton4.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jButton4.setText("Submit");
        jButton4.addActionListener(this::jButton4ActionPerformed);
        jLabel7.setFont(new java.awt.Font("Segoe UI Emoji", 0, 18)); // NOI18N
        jLabel7.setText("Language :");
        jComboBox1.setFont(new java.awt.Font("Segoe UI Semibold", 0, 18)); // NOI18N
        jComboBox1.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "C", "C++",
"JAVA" }));
        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```
            .addComponent(jLabel2)
            .addGap(157, 157, 157)
            .addComponent(jLabel3)
            .addGap(136, 136, 136)
            .addComponent(jLabel4)
            .addGap(90, 90, 90))
          .addGroup(layout.createSequentialGroup()
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
              .addGroup(layout.createSequentialGroup()
                .addGap(34, 34, 34)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                          .addComponent(jLabel9)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED))
                        .addGroup(layout.createSequentialGroup()
                          .addComponent(jLabel10)
                          .addGap(19, 19, 19)))

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                        .addComponent(jScrollPane4, javax.swing.GroupLayout.DEFAULT_SIZE,
150, Short.MAX_VALUE)
                        .addComponent(jScrollPane5))
                      .addGap(12, 12, 12))
                    .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addComponent(jButton3)
                        .addComponent(jButton4)
                        .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addComponent(jLabel6)
                            .addComponent(jLabel5)
                            .addComponent(jLabel7))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                                .addComponent(jButton2)
                                .addComponent(jButton1)
                                .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE, 127, javax.swing.GroupLayout.PREFERRED_SIZE))))
                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)))
                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 226,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 226,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(jScrollPane3, javax.swing.GroupLayout.PREFERRED_SIZE, 226,
javax.swing.GroupLayout.PREFERRED_SIZE))
              .addGroup(layout.createSequentialGroup()
                .addGap(443, 443, 443)
                .addComponent(jLabel11))
              .addGroup(layout.createSequentialGroup()
                .addGap(406, 406, 406)
                .addComponent(jLabel1)))
            .addContainerGap(49, Short.MAX_VALUE))
      );
      layout.setVerticalGroup(
         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
         .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jLabel1)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
               .addComponent(jLabel2)
               .addComponent(jLabel3)
               .addComponent(jLabel4))
            .addGap(1, 1, 1)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
               .addComponent(jScrollPane3, javax.swing.GroupLayout.DEFAULT_SIZE, 402,
Short.MAX_VALUE)
               .addComponent(jScrollPane2)
```

```
            .addComponent(jScrollPane1)
            .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel5)
                .addComponent(jButton2))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel6)
                .addComponent(jButton1))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(jLabel7, javax.swing.GroupLayout.PREFERRED_SIZE, 34,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGroup(layout.createSequentialGroup()
                  .addComponent(jComboBox1, javax.swing.GroupLayout.PREFERRED_SIZE, 30,
javax.swing.GroupLayout.PREFERRED_SIZE)
                  .addGap(4, 4, 4)))
            .addGap(18, 18, 18)
            .addComponent(jButton4)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jButton3)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                    .addComponent(jLabel9)
                    .addComponent(jScrollPane4, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                  .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                  .addComponent(jLabel10))
                .addComponent(jScrollPane5, javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))))
```

```
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 48,
Short.MAX_VALUE)
            .addComponent(jLabel11)
            .addContainerGap())
        );
        pack();
    }// </editor-fold>
    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        //output
        File file;
        System.out.println("zobair");
        JFrame frame = new JFrame("Hello");
        JFileChooser fc = new JFileChooser();
        int returnVal = fc.showOpenDialog(frame);
        if(returnVal == JFileChooser.APPROVE_OPTION){
            String path = fc.getSelectedFile().getAbsolutePath();
            file = fc.getSelectedFile();
            System.out.println(file);

            try{
                Scanner scanner;
                scanner = new Scanner(file);
                String input = "";
                while (scanner.hasNextLine()) {
                input = input+scanner.nextLine()+"\n";
                }
                scanner.close();
                jTextArea2.setText(input);
            } catch (FileNotFoundException e){
            }
        }
        else{
            System.out.println("Open command cancelled by user");
        }
        System.out.println(returnVal);
    }  //output

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        // code
```

```
    File file;
    System.out.println("zobair");
    JFrame frame = new JFrame("Hello");
    JFileChooser fc = new JFileChooser();
    int returnVal = fc.showOpenDialog(frame);
    if(returnVal == JFileChooser.APPROVE_OPTION){
       String path = fc.getSelectedFile().getAbsolutePath();
       file = fc.getSelectedFile();
       System.out.println(file);
       try{
          Scanner scanner;
          scanner = new Scanner(file);
          String input = "";

          while (scanner.hasNextLine()) {
          input = input+scanner.nextLine()+"\n";
          }
          scanner.close();
          jTextArea1.setText(input);
       } catch (FileNotFoundException e){
       }
    }
    else{
       System.out.println("Open command cancelled by user");
    }
    System.out.println(returnVal);
}  // code

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    //clear
    jTextArea1.setText(null);
    jTextArea2.setText(null);
    jTextArea3.setText(null);
    jTextPane2.setText(null);
    jTextPane1.setText(null);
}  //clear
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {            //submit
    // TODO add your handling code here:
    //submit
    String res = null, txt;
```

```
        //write data from jTaxtArea1 to submitted.txt
        try {
            try(BufferedWriter ofile = new BufferedWriter(new
FileWriter("D:\\STUDY\\3rd_semister\\Run\\submitted.txt"))) {
                jTextArea1.write(ofile);
                ofile.flush();
                ofile.close();
            }
        } catch (IOException ex) {
            Logger.getLogger(P1.class.getName()).log(Level.SEVERE, null, ex);
        }
        //CAll bat file..
        double startTime = System.currentTimeMillis();
        System.out.print(startTime);
        //startTime = clock();
        try {
            Process p = Runtime.getRuntime().exec(new
String[]{"D:\\STUDY\\3rd_semister\\Run\\01.bat"});
            p.waitFor();
        } catch (IOException er) {
            System.out.println("Error!222");
        } catch (InterruptedException er) {
        }
        double endTime = System.currentTimeMillis();
        System.out.print(endTime);
        //endTime = clock();
        //read output
        try {
            try (BufferedReader ifile = new BufferedReader(new
FileReader("D:\\STUDY\\3rd_semister\\Run\\verdict.txt"))) {
                res = ifile.readLine();
            }
        } catch (IOException er) {
        }
    //Print output of user in jTextArea3..
    int flag2;
        StringBuilder buffer2 = new StringBuilder("");
        FileInputStream inputStream2 = null;
        try {
            inputStream2 = new FileInputStream(new
File("D:\\\\STUDY\\\\3rd_semister\\\\Run\\\\output.txt"));
```

```
        } catch (FileNotFoundException ex) {
            Logger.getLogger(P1.class.getName()).log(Level.SEVERE, null, ex);
        }
        try {
            while( (flag2 = inputStream2.read()) != -1)
                buffer2.append((char) flag2);
        } catch (IOException ex) {
            Logger.getLogger(P1.class.getName()).log(Level.SEVERE, null, ex);
        }
        try {
            inputStream2.close();
        } catch (IOException ex) {
            Logger.getLogger(P1.class.getName()).log(Level.SEVERE, null, ex);
        }
        String abc;
        abc = buffer2.toString();
        jTextArea3.setText(abc);

        //Show verdict & time
        jTextPane2.setText((endTime-startTime)+  "  ms");
        jTextPane1.setText(res);
        /**Delete outputs from folder
        try {
            Process p;
            p = Runtime.getRuntime().exec("D:\\STUDY\\3rd_semister\\Run\\del.batch");
            p.waitFor();
        } catch (IOException er) {
            System.out.println("Error!0101");
        } catch (InterruptedException er) {
        }*/
    }
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
```

```
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException | InstantiationException | IllegalAccessException |
javax.swing.UnsupportedLookAndFeelException ex) {
        java.util.logging.Logger.getLogger(P1.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    }
    //</editor-fold>

    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(() -> {
        new P1().setVisible(true);
    });
  }
   // Variables declaration - do not modify
  private javax.swing.JButton jButton1;
  private javax.swing.JButton jButton2;
  private javax.swing.JButton jButton3;
  private javax.swing.JButton jButton4;
  private javax.swing.JComboBox jComboBox1;
  private javax.swing.JLabel jLabel1;
  private javax.swing.JLabel jLabel10;
  private javax.swing.JLabel jLabel11;
  private javax.swing.JLabel jLabel2;
  private javax.swing.JLabel jLabel3;
  private javax.swing.JLabel jLabel4;
  private javax.swing.JLabel jLabel5;
  private javax.swing.JLabel jLabel6;
  private javax.swing.JLabel jLabel7;
  private javax.swing.JLabel jLabel9;
  private javax.swing.JScrollPane jScrollPane1;
  private javax.swing.JScrollPane jScrollPane2;
  private javax.swing.JScrollPane jScrollPane3;
  private javax.swing.JScrollPane jScrollPane4;
  private javax.swing.JScrollPane jScrollPane5;
  private javax.swing.JTextArea jTextArea1;
```

```
    private javax.swing.JTextArea jTextArea2;
    private javax.swing.JTextArea jTextArea3;
    private javax.swing.JTextPane jTextPane1;
    private javax.swing.JTextPane jTextPane2;
    // End of variables declaration
}
```

Evaluator:

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int t =6;
    FILE *in,*jg, *ou;

    in = fopen("output.txt","r");

    jg=fopen("judge.txt", "r");

    ou = fopen("verdict.txt", "w");

    char si[1000000], sj[1000000];

    while(t--)
    {
        fscanf(jg, "%s", sj);
//        getchar();
        fscanf(in, "%s", si);
  //      getchar();

    //    printf("t = %d\njudge = %s\ninput = %s\n",t,  sj, si);

        if (strcmp(si, sj) != 0)
        {
            fprintf(ou, "Wrong Answer\n");
            fclose(in);
            fclose(jg);
            fclose(ou);
            return 0;
```

```
    }
  }
      fprintf(ou, "Accepted\n");
      fclose(in);
      fclose(jg);
      fclose(ou);
  return 0;
}
```

Output Generator:

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
  int fl = 0;
  string str, s;
  ifstream in;
  ofstream out;
  in.open("submitted.txt");
  out.open("code.cpp");
  while(!in.eof())
  {
    getline(in, str);

    if (fl != 2)
    {
      stringstream strm;

      strm << str;

      while(strm >> s)
      {
        out << s << " ";
        if (s == "main" || s == "main(" || s == "main()" || s == "main(){")
        {
          fl = 1;
          continue;
        }

        if (fl == 1)
        {
```

```
                if (s == "{")
                {
                    out << "freopen(\"input.txt\", \"r\", stdin); \n freopen(\"output.txt\", \"w\",
stdout);" << endl;
                    fl = 2;
                }
            }
        }
        out << endl;
        continue;
    }
    out<<str<<endl;
  }
  return 0;
}
```

Usefulness: By this app a user can easily check that his code is correct or not and the time complexity of his code. And is don't have to get wrong answer in the main judgment.

Limitations: Here is no server, so by this app no one can take an onsite contest.

Discussion: In the next version of this app some feature will be added like it can measure the runtime error, compiler error, etc. now, this app only can compile C, C++ and JAVA code. But in the next version it will compile Pascal, Python etc.