IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 4, November 2016)

Algorithm for Grading a Computer Program using the Inverse Function

Harendra Kumar[1]
[1]*Assistant Professor*
SIET Sikar, Rajasthan

Shivraj Sharma[2]
[2]*PG Scholar*
SIET Sikar, Rajasthan

Suneel Kumar[3]
[3]*Assistant Professor*
SIET Sikar, Rajasthan

***Abstract*— Computer Programs are written to perform a particular task. We design various computer programs to simplify our job and make our work flawless.So we must have a way to evaluate the correctness of the program written by us. In this research we have evolved an algorithm to grade computer programs for the efficiency. This paper is about students who want to learn programming languages from basics or advanced and have to evaluate their programming skills for further development.**

In our proposed work we have developed auto-grading tool that can grade C/C++ Programs. This is an offline tool which accepts a program submitted by a student and grades it using static and dynamic analysis.

***Keywords*— Image Encryption, Image Decryption, Image Passwords.**

I. Introduction

Since 1960, numerous domestic and foreign experts and researchers have devoted to the exploration of insightful scoring technology in view of C dialect programming; likewise there have been different programmed scoring frameworks. The present scoring strategies can be separated into three classes: First, programming quality metrics technique; Second, dynamic analysis strategy; third is the correlation strategy for source code. To a specific degree the metrics parameters connected in programming quality metrics can show the structure of the program, I/O mode, the elements data flow and control flow-sensitive, yet be poor in the comprehension of the source code structure, and not investigate program semantics. It's hard to get satisfactory result for just utilizing the technology that is utilized to score program of the hopefuls. The analysis and correlation technique for source code is exceptionally appropriate to score naturally program of the applicants by PCs, yet it's is somewhat hard to usage. Presently, at home and abroad, the researches here are very few, and genuine down to earth frameworks are few, the programmed scoring frameworks in light of the entire analysis of source code are less. Dynamic analysis strategy is the most immediate and correct to decide the practical proportionality of program, yet can't manage the not performable programming questions for, for example, grammar or semantic mistakes.

The smart scoring framework show has been at first settled for the subjective questions(C programming dialect programming and fill in the clear inquiries) in the C dialect current examination framework, yet there are still a few deficiencies in the part of its source code analysis and correlation. These frameworks are insufficient human wisely for assessment of scholastic accomplishment and the check of information substance, and exist deformities of proper scoring as per sensible strides of reply.

After broad analysis and research, in view of the fundamental thoughts and techniques for particular standards of savvy scoring strategy for the semantic comparability, consolidated with investigating source code by utilizing the dynamic and static semantic comprehension technique, the astute scoring technique in light of semantic likeness of the program is proposed. Steps are as per the following:

Comes about intercomparison. Check the consequence of the hopefuls program whether the document with the standard answer was steady.

Dynamic semantic analysis. Whether the program is right is controlled by the aftereffects of running system. On the off chance that right, the scoring is resolved whether the program structure meets the importance of the inquiries.

Static semantic analysis. The hopefuls program is scored by coordinating and getting semantic similitude between the competitors program and the template program (the answer program). The higher the semantic similitude is, the higher the score of the hopefuls program is. The technique is called as the semantic comparability scoring strategy in view of the static semantic comprehension in the paper.

The above scoring strategy still can't understand to score by steps the same as the manual scoring, however contrasted with past scoring as per the outcomes, it is trusted that the framework still has a positive significance. The test inquiries are some littler program, so the scoring can be more noteworthy degree exactness

II. Related Work

Fengxia Yang [1], Guoxiang Liu [1] proposed new programmed scoring technique which consolidates the key proclamation focuses with the lexical or linguistic blunder examination is proposed for programming test through the recreation of manual scoring. Through the trials, the proposed technique in a general sense compares with the manner of thinking of manual scoring. Additionally, the likelihood circulation of the programmed scoring result is like the one of manual scoring result. It has extraordinary centrality for acknowledging paperless testA new programmed scoring strategy which consolidates the key proclamation focuses with the lexical or syntactic mistake investigation is proposed for the C programming test for the full-time understudies .Through countless, the accuracy was broke down and it can gauge the candidates' test well. The scoring result from the programmed scoring above is nearer to the one from manual scoring than the ordinary scoring by the immediate results. It has extraordinary hugeness for acknowledging paperless test.

Wen Xiangmin, Hu Lin [2] , advises that as a few issues exists in C language programming wise scoring innovation, similar to its insight is poor and its decide is not precisely. Another strategy in light of semantic likeness score is composed. The static brushed with element semantic comprehension are utilized as a part of it, so the score are more accommodating, all the more near the results of manual scoring, likewise the assurance exactness is moved forward. This paper shows a generally new canny scoring of C-language programming questions. By utilizing blend mode amongst static and element semantic understanding, coordinating degree scoring on semantic comparability, scale, structure , sentence between the candidates program subsequent to institutionalizing and answer format program ,enhancing the scoring knowledge of the examination framework, the strategy has been at first accomplished by outlining code. By testing, scoring result is more humanness, nearer to the manual scoring result. It makes the candidates results all the more reasonable, more sensible. To a specific degree, it diminishes work force of educators different parts of subjective considers the examination. It can sort out examinations all the more rapidly, equitably and securely, free instructors from the substantial paper, checking and results investigation.

Xiao Zhao Liu Xuefeng Hou Yumo [3] , recommended that the programs having sentence structure blunders submitted to some programming examination programmed scoring framework would be just given zero point, the result is not matched with the propensity for counterfeit test paper stamping. With a specific end goal to tackle this issue, a programmed reviewing calculation which in light of the hypothesis of compiler hypothesis is given. The calculation gave a strategy which consolidated element stamping and static checking, and the static checking accomplished with coordinated key information focuses. As per the test, the calculation has precision of commenting and some profitable spaces of use.

Yuki Akahane, Hiroki Kitaya, and Ushio Inoue,[4] This paper displays an electronic programmed scoring framework for Java programming assignments, and reports assessment results in a real programming course. The framework gets Java application programs put together by understudies and returns the test outcomes promptly. The test comprises of compiler check, JUnit test, and result test. The result test is exceptionally helpful for assignments in rudimentary programming courses, in light of the fact that a run of the mill program is made out of just a principle technique that peruses/composes information from/to the standard info/yield gadgets. The framework was utilized and assessed as a part of a real course of our college. We affirmed that the framework is extremely useful for understudies to enhance their programming abilities. Particularly, numerous understudies saw and remedied their errors by rehashing accommodation of their programs a few times.

III. Proposed Concept

In the proposed concept we have designed a framework in Eclipse that will proceed in the following steps to grade the programs submitted by the user:

1. Student/User will submit a C Program to this tool and press submit button.
2. First, this submitted program will be compiled; If compilation is successful will continue grading process, otherwise compilation errors are shown to student.

3. After successful compilation, this tool will execute the user program and test cases are run on this program and output of all these test cases will be stored in a database.

4. A reference/model program for the same task as of user's program is taken. The reference program has high degree of efficiency has to be compiled. The same test cases have to be run for this model program and the results of these test cases are stored on the database prior to grade the user submitted program.

5. Output of both programs will be matched; if output is same marking or grading will be done otherwise zero will be given.

6. Grading Report will be displayed for the user/student.

The auto-grader captures the respective blocks (functions) of student program code one by one and replaces with the appropriate function/block in the Sample/reference program/solution.

The modified program is then executed for 4 to 6 test cases (each at least 5times).

Marks are awarded for a test case only if the output matches with the actual output for all 5 times for that test case.

This procedure is followed for all the blocks.

Each block carries 1 mark. If there are 5 test cases, then every correct output of a block for that particular test case will be awarded 0.2 marks

Testing Block 2 :

Testcase = '../1week/solutions/2015-02-12/testcases/testcase0', Score = 0.20

Testcase = '../7week/solutions/2015-02-12/testcases/testcase1', Score = 0.20

Testcase = '../7week/solutions/2015-02-12/testcases/testcase2', Score = 0.20

Testcase = '../7week/solutions/2015-02-12/testcases/testcase3', Score = 0.20

Testcase = '../7week/solutions/2015-02-12/testcases/testcase4', Score = 0.20

Block 2 Score = 1.00

Testing Block 3 :

Testcase = '../7week/solutions/2015-02-12/testcases/testcase0', Score = 0.20

Testcase = '../7week/solutions/2015-02-12/testcases/testcase1', Score = 0.00

Testcase = '../7week/solutions/2015-02-12/testcases/testcase2', Score = 0.20

Testcase = '../7week/solutions/2015-02-12/testcases/testcase3', Score = 0.20

Testcase = '../7week/solutions/2015-02-12/testcases/testcase4', Score = 0.20

Block 3 Score = 0.80

Testing Block 4 :

Empty block 4. Skipping evaluation.

Block 4 Score = 0.00

Algorithm for Evaluation: Inverse Function

A function which follows unique value property or one -to-one property has their inverse functions.

1. Find the sum of two numbers X and Y.

   Student Solution: Z= X+Y
   F(Z)= X+Y
   X=2;
   Y=3;
   F(Z)= 2+3
   = 5

   Verification program: Inverse Function

   F-1 (Z)= F(Z)- F(X)or F(Y)
   =5-2 or 5-3
   =3 or 2

We used student output F(Z) and one of the input value if we get back another input value of program correct then student program is correct otherwise wrong.

2. Find the difference of two numbers 8 and 4.

Student Solution: x=8;

y=4;
F(z) = x-y;
F(z) = 8-4;
= 4;

Verification program: Inverse Function
F-1 (z) = F(z) + y
= 4+4
=8

When we use student output and add it to the input value y we get back x. If x comes out correct then student program is correct.

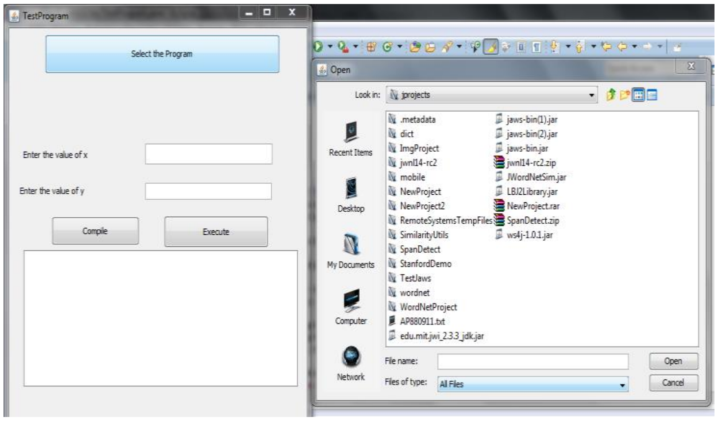
3. Find the product of two numbers.

Student Solution: X=5;
Y=6;
F(Z) = X * Y
= 5*6;
= 30

Verification program: Inverse function
F-1(Z) = F(Z) / X or Y
= 30/5 or 6
= 6, 5
When we divide the student output result by one of the input value we get back the original input values if input values of program comes out correct then student o/p is correct.

Visual Execution of the proposed concept

Step 1: Click on the “Select the Program “button ,

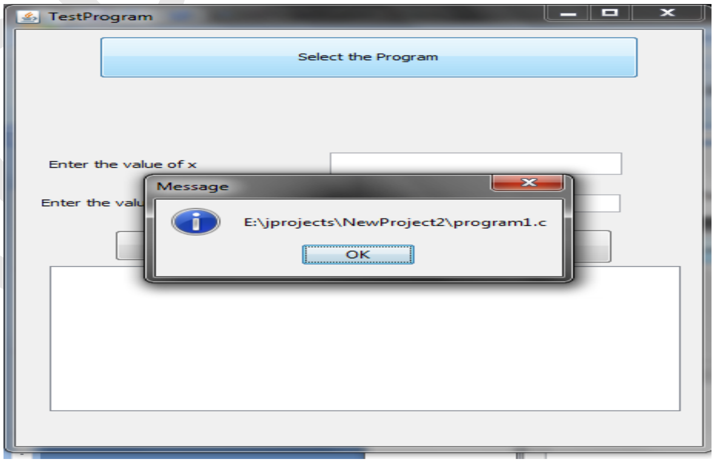


Look in: jprojects

Files of type: All Files

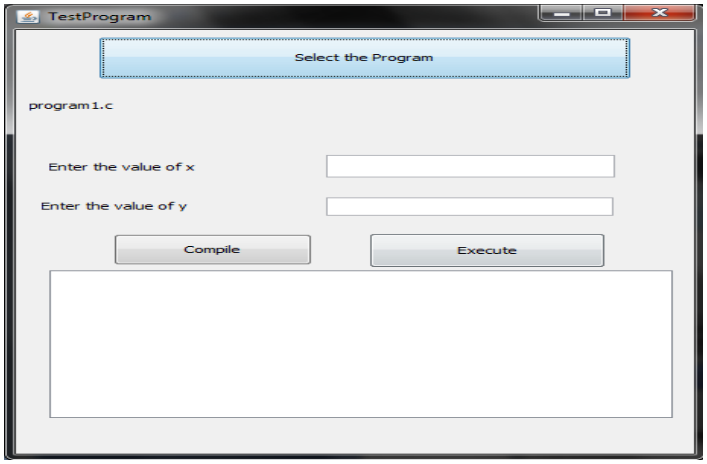
Figure1 : Select the Program

Step 2: Program selected for compilation



E:\jprojects\NewProject2\program1.c

Figure 2: Alert for Program Selection



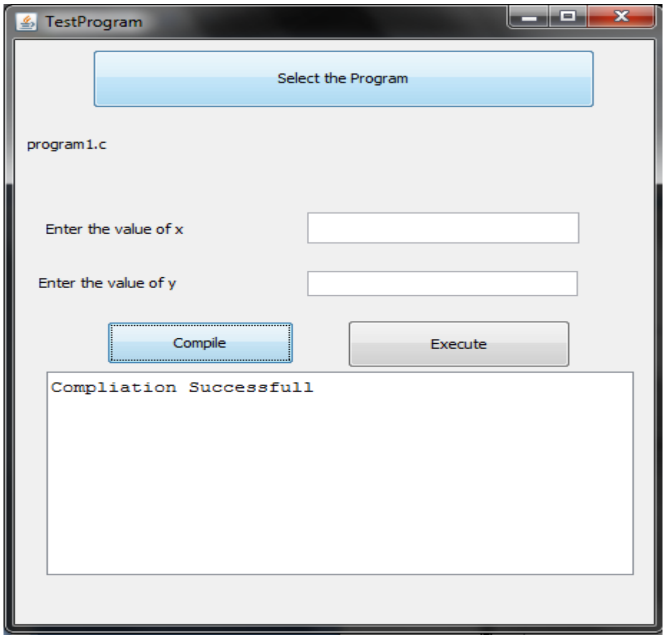
program1.c

Figure 3: Selected Program Name on Label

Step 3: Click on the compile button

IJDACR
International Journal Of Digital Application & Contemporary Research

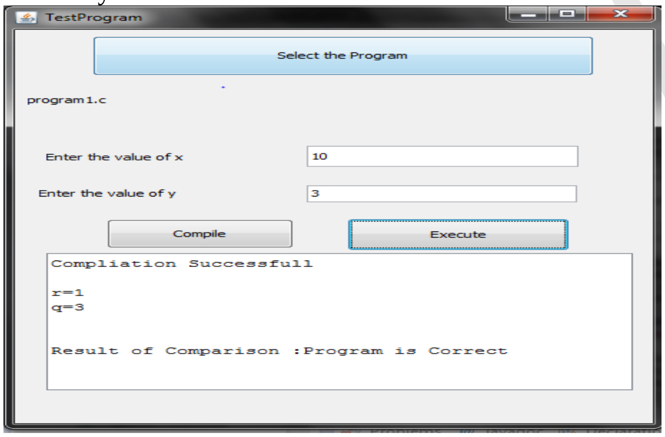
International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 5, Issue 4, November 2016)



```
Compliation Successfull
```

Figure 4: Compilation using “C/C++” compiler

Step 4: Click on the execute button after giving the values of x and y



Enter the value of x: 10

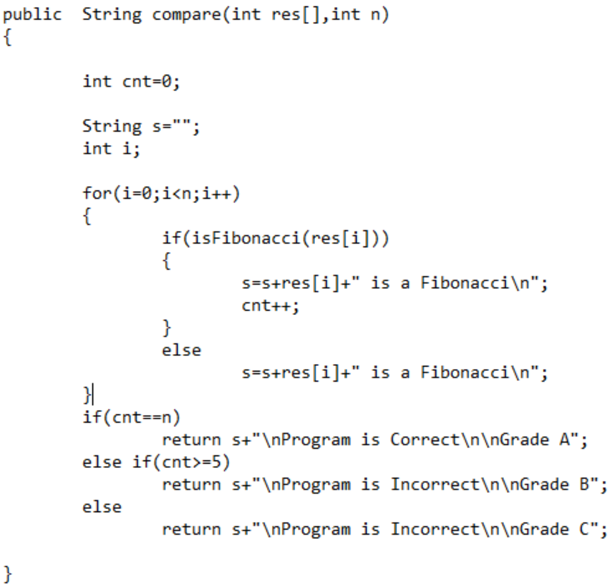
Enter the value of y: 3

```
Compliation Successfull

r=1
q=3

Result of Comparison :Program is Correct
```

Figure 5: Result of Comparison



```
public  String compare(int res[],int n)
{

        int cnt=0;

        String s="";
        int i;

        for(i=0;i<n;i++)
        {
                if(isFibonacci(res[i]))
                {
                        s=s+res[i]+" is a Fibonacci\n";
                        cnt++;
                }
                else
                        s=s+res[i]+" is a Fibonacci\n";
        }
        if(cnt==n)
                return s+"\nProgram is Correct\n\nGrade A";
        else if(cnt>=5)
                return s+"\nProgram is Incorrect\n\nGrade B";
        else
                return s+"\nProgram is Incorrect\n\nGrade C";

}
```

Figure 6: Grading Evaluation Program Code.

IV. CONCLUSIONS

Each person has its own programming strategy but still there is always a requirement to find the methods the methods which perform the task in the most efficient way. To accomplish this objective we must have a tool to grade a computer program for the required degree of efficiency. Such grader which we have proposed will let the learning students and trainer evaluate their code and performance and this will help the programmer to improvise his or her code so that a high level of efficiency can be achieved

ACKNOWLEDGMENT

It is my pleasure and privilege to acknowledge and express my deep sense of gratitude to my teacher and guide, Mr. Rahul Singh, (Assistant Professor), Department of Computer Science & Engineering, Faculty of Engineering, Shekawati institute of Engineering & Technology, Sikar, who inspired and initiated me to prepare this research despite his busy academic schedule. He has always been kind enough to spare his valuable time and thought in giving necessary guidance. His rich and varied experience as an academician immensely helped me in understanding this topic clearly

I express my sincere thanks to Mr. Vaibhav Doshi Department of Computer Science& Engineering, Shekawati institute of Engineering & Technology, Sikar , for his encouragement.

REFERENCES

[1] Fengxia Yang , guoxiang Liu ,"Study and application of automatic scoring technology in C programming test, IEEE, 2012.

[2] Wen Xiangmin, Hu Lin,"The research of C language programming intelligent scoring technology based on the semantic similarity", IEEE, 2012.

[3] Xiao Zhao Liu Xuefeng Hou Yumo,"Research and Implementation of Automatic Scoring System about Programming", IEEE, 2012.

[4] Yuki Akahane, Hiroki Kitaya, and Ushio Inoue,"Design and Evaluation of Automated Scoring Java Programming Assignments", IEEE, 2015.

.