Similarity detection tools are commonly used by teacher ,guide to detect and prevent student cheating in academic assignments . To create environment for student to solve assignments using their own ideas.

Systems identifying similarity in programming assignments are generally classified as either attribute-based or structure-based systems. Attribute-based method is used to detect lexical changes using program attribute and structure-based method analysis complete structure of program to detect structure similarities.

Introduction :

Plagiarism is an attempt to pass off someone's work, in a whole or part, as his/her own work without giving credit. So most of students who copy materials from different book, materials without citing references. In lexical type small changes can be done in programming like adding or deleting comments, changing variables name etc. while in Structural one can change structure of program without altering its significance. Similarity can be detected automatically and manually also .but an effort and time required to detect similarity is more in manual as compared to automatic detection.

There are three different classes of similarity detection methods: quiz methods, writing style methods, and comparison methods with original sources. So here we are dealing with comparison methods with original source.

There are Existing system presents which is used to find similarity in code. Those are

Measure Of Software Similarity(MOSS), JPlag , CodeMatch, Copy/Paste Detector(CPD).

PROPOSED SYSTEM :

Source code implemented by students are characterized by common traits. So plagiarism can be detect by focusing on lexical changes because sometimes structural changes are not come in to picture. In short programming code it might be useless to compare source code structure.

one of the example is CAPlag (Computing Assignment Plagiarism) use to check plagiarism in java programs. It consists of two phases. First phase consists of fast screening process that compare program profile and In second phase algorithm goes in detail and compare code.

1. First phase: Attribute-based comparison

In phase one plagiarism detection is done on program metrics .This metrics are divided into three categories :

1. programming layout metrics
2. programming style metrics
3. programming structure metrics

So in above categories it will focus on keywords, addition and deletion comments, renaming variables, counting open and close braces, operators, character length etc.

Using above measures we find similarity in program using Weighted Mean formula for layout, style, structure of each program . Then we compare two program based on weighted Mean.

1. Second phase: Structure-based comparison

In Structure –based comparison similarity has quantitative and qualitative aspects. Quantitative aspect show degree of similarity in files. Qualitative answer shows where they are similar and where they are different.

Conclusion:

Attribute- based method can be useful for small and medium size programs.

Structure-based method is useful for large size programs.