OBSERVATIONS ON PLAGIARISM IN PROGRAMMING COURSES

Branko Kaučič, Dejan Sraka, Maja Ramšak
Department of mathematics and computer science, Faculty of Education, University of Ljubljana
Kardeljeva ploščad 16, Ljubljana, Slovenia

Marjan Krašna
Faculty of Arts, University of Maribor, Koroška cesta 16, Maribor, Slovenia

Keywords: Plagiarism, Source code, Teaching programming, Programming assignments.

Abstract: Plagiarism is a well known problem of today's society. Widespread of the internet, ease of data exchange, Bologna reforms and individual circumstances influence on students to resort to plagiarism. Many courses in computer science where students have programming assignments suffer from so called source code plagiarism. Beside the internet, most common origins of solved assignments are fellow students from the same or the previous generation.

In this paper the source code plagiarism is discussed. Main results from observing the plagiarism in programming assignments are given showing to which extent students plagiarize.

1 INTRODUCTION

Reproducing someone’s work without acknowledging the source is known as plagiarism. Research indicates that plagiarism is a significant problem in today’s institutions of higher education (Austin, 1999; Baggaley, 2005; Bennett, 2005; Hammond, 2004; Moussiades, 2005). In most cases students copy and paste without proper citing. In many cases they are even not aware of committing a fraud, because they are not aware how to use other resources in their own work.

At present, educational institutions “fight” for an increased number of students while reducing the number of contact hours and preserving same staff number. The opportunities for teachers to identify the frauds and the students that need additional help (Joy, 1999; Sraka, 2009) decreased. Many of students end courses with insufficient knowledge.

Although the plagiarism is often committed with written text, it is also regularly committed with the software code. In most computer science courses, programming assignments where students have to write a piece or complete source code by given specifications, are part of course obligations. Since programming skills are not among easiest, and mastering them requires a lot of practice and understanding, some students resort to academic dishonesty. At Faculty of Education at University of Ljubljana we started a project of studying plagiarism. Initially, we restricted the research on source code plagiarism in computer science courses.

The organization of the paper is the following. Section 2 presents the problem of plagiarism. In section 3 the main results of observing the plagiarism at specific programming course are presented, and in section 4 the paper is concluded.

2 PRELIMINARIES

The term “plagiarism” has many definitions, sharing the same idea as “a piece of writing that has been copied from someone else and is presented as being your own work” (www.websters-online-dictionary.org). Some of the reasons why students plagiarize can be found in the following categories (Bennett, 2005): means and opportunity, personal traits, and individual circumstances.

Kaučič B., Sraka D., Ramšak M. and Krašna M. (2010).
OBSERVATIONS ON PLAGIARISM IN PROGRAMMING COURSES.
In *Proceedings of the 2nd International Conference on Computer Supported Education*, pages 181-184
DOI: 10.5220/0002800701810184

2.1 Plagiarism in Programming Courses

Plagiarism is a common problem in computer science courses. In many cases, the completion of programming assignments is a part of the course requirements. In (Parker, 1989), source code plagiarism is defined as “a program which has been produced from another program with a small number of routine transformations.” Changes can vary from copy and pasting small amounts of program source code to copy and pasting large chunks of source code and masking everything with different disguising techniques. Possible modifications range in sophistication levels ranging from 1 to 6 (Faidhi, 1987). For the educational purposes it is more important to identify the changes on lower levels. Based on experiences we could also add level 0 at which no changes are made to copied source code.

In programming courses there are usually two sources of solved assignments: the internet and other students. The second source, other students from current and previous generations, is much more frequent. Regardless of the source, detecting plagiarized assignment can be difficult task. At high number of students and assignments this is sometimes even impossible; despite the effort the assistant cannot thoroughly check all source codes. Therefore, plagiarism is quite often not detected or accidentally.

Since there is usually more than just a few assignments, programming courses are in a desperate need for an automated tool – the plagiarism detection system. There are various systems to detect plagiarism in source codes (Ahtiainen, 2006; Clough, 2000), some are web based applications (Bowyer, 1999; Prechlet, 2000).

2.2 WMajorClust Algorithm

Detection systems usually report which source codes are similar to other source codes. Observing solely these values one can see which students plagiarized but it is not obviously evident which students also participated in this and share the same code. To overcome this, we can use the WMajorClust algorithm (Niggemann, 2001) to perform clustering of plagiarized source codes and authors, respectively. In similar fashion the algorithm was used as a second phase of PDetect system (Moussiades, 2005). Note, that WMajorClust uses a parameter “cut-off criterion” which represents the minimum similarity between two source codes to include them in clustering.

3 OBSERVATIONS ON PLAGIARISM

Students that will become teachers of mathematics and computer science in elementary and some secondary schools start to learn programming in two courses: Computer practice and Programming. Some of the obligations in Programming are programming home-works and seminar works for Pascal, C and PHP. In this paper we observed the occurrence of plagiarism in Pascal assignments, which for specific study year consisted of:

2007/2008:
Homework 1: sets (78HW1)
Homework 2: arrays (78HW2)
Homework 3: recursion (78HW3)
Homework 4: records (78HW4)
Homework 5: files (78HW5)
Homework 6: pointers (78HW6)
2008/2009:
Homework 1: sets (89HW1)
Homework 2: arrays (89HW2)
Homework 3: subprograms (89HW3)
Homework 4: strings and records (89HW4)
Homework 5: recursion (89HW5)
Homework 6: files (89HW6)
Homework 7: pointers (89HW7)
Seminar work: (89SW)
2009/2010:
Homework 1: renewal (91HW1)
Homework 2: sets (91HW2)

Assignments differ each study year, and all students always get identical assignments.

In total, we observed 102 students, 85 females and 17 males. Table 1 shows numbers of them in different study years. Some students appear in more than one study year.

Table 1: Number of students for different study years.

study year	female	%	male	%	students
07/08	44	88,0	6	12,0	50
08/09	36	85,7	6	14,3	42
09/10	21	72,4	8	27,6	29

Students’ assignments were sent to MOSS detection system (Bowyer, 1999). Its quality is positively observed in (Culwin, 2001) and (Zeidman, 2006). Results of the similarity were then used as the input to WMajorClust, cut-off criterion was set to 50. Detailed view on plagiarism for specific assignments shows table 2. It contains the number of students that submitted assignments, number of students that were detected as possible plagiarists, percentage of them against number of submissions. Last three columns show number of clusters, maximal number of students in the biggest cluster and average number of students in clusters.

Combined view in the number of plagiarized assignments for each study year shows first four columns in table 3. Percentage view over study years is visualized in figure 1. Column "max" shows the maximal number of assignments that were plagiarized by the same student, the row "continuous" shows how many students continued to plagiarize after their first plagiarized assignment, and the last row shows number of different students that plagiarized in specific study year.

Table 2: Plagiarism by individual assignments.

assignment	submitted	plagiarist	%	clusters	max	avg
78HW1	49	14	28,6	5	4	2,8
78HW2	48	21	43,8	10	3	2,1
78HW3	46	10	21,7	4	3	2,5
78HW4	45	20	44,4	9	3	2,2
78HW5	44	16	36,4	5	6	3,2
78HW6	39	17	43,6	7	4	2,4
89HW1	42	14	33,3	6	4	2,3
89HW2	42	19	45,2	7	4	2,7
89HW3	37	13	35,1	6	3	2,2
89HW4	50	26	52,0	12	6	2,2
89HW5	33	12	36,4	4	4	3,0
89HW6	33	22	66,7	6	9	3,7
89HW7	32	25	78,1	9	5	2,8
89SW	25	18	72,0	6	5	3,0
91HW1	28	2	7,1	1	2	2,0
91HW2	28	5	17,9	2	3	2,5

Table 3: Plagiarism over study years.

study year	submitted	plagiarized	%	max	continuous	plagiarists
07/08	271	98	36,2	6	7	34
08/09	294	149	50,7	7	2	38
09/10	56	7	12,5	2	2	5

It can be seen that in 2007/2008 one student plagiarized all (6) assignments, in 2008/2009 one student plagiarized all assignments except one (7) and in 2009/2010 two students plagiarized both assignments so far.

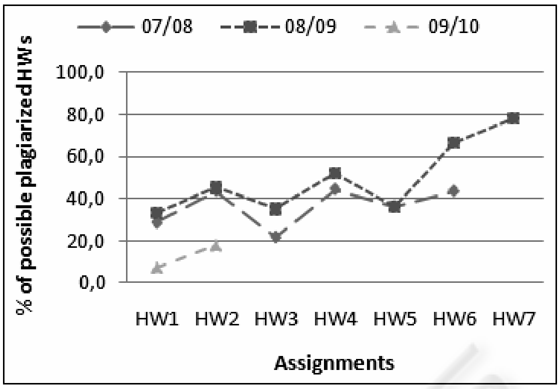

Figure 1: Plagiarism over study years.

As can be seen from the graph on figure 1, study year 2008/2009 resulted in high percentage of plagiarism which was even increasing over the assignments. This year students were warned about the plagiarism on lectures and laboratory exercises, educated about the fraud and students were warned that their assignments will undergo through detection system. Positive results are evident on the graph; only one student plagiarized so far.

Detailed statistics of how many students plagiarized one, two, etc. assignments shows table 4. In 2007/08 students mostly plagiarized one assignment, and in 2008/2009 students mostly plagiarized more than half of all assignments. Although the grades of final exams are not given in this paper, we can state that plagiarism reflected in grades. Higher plagiarism ratio resulted in worse grades and failures at the exams.

Table 4: How many assignments students plagiarized over study years.

plagiarized assignments	07/08	08/09	09/10
1	11	5	3
2	5	3	2
3	6	8	/
4	4	7	/
5	5	7	/
6	3	5	/
7	/	3	/

Table 5 shows the number of students that plagiarized grouped by the gender. Columns with percentages exhibit percentage of students against total number of students (from table 2), for males and females, respectively. Average percentage for females is 58,7% and 59,7% for males from which we can conclude that there is no significant difference in plagiarism by the gender.

Table 5: Plagiarism by gender over study years.

study year	females	%	males	%	plagiarists
07/08	30	68,2	4	66,7	34
08/09	32	88,9	6	100	38
09/10	4	19,0	1	12,5	5

4 CONCLUSIONS

Plagiarism is a common problem in programming courses, especially in today's copy-paste generation. Its complexity demands serious approach at solving it: by using appropriate detection systems, proper regulation, proper assignments and education of students about it. Several tips for practitioners and for students how to deal with the plagiarism are given in (Austin, 1999; Schiller, 2005).

The success of decreasing the plagiarism problem heavily depends on formal regulations, rules and procedures. Secondary aim of a proper regulation is also to protect and guide the teachers when accusation is started, and the students against injustice accusation and sanctions. How delicate cases can occur, can be seen in (Baggaley, 2005).

Reducing the plagiarism significantly depends also from the teachers. Efficient advice is to choose assignments that allow several interpretations and reduce the probability to obtain identical or semi-identical results. Each study year teachers should also change assignments and prevent reusing of source code between generations of students.

Important factor in reducing plagiarism among students is in educating about it. Teachers and students have to be educated about the importance of authorship, intellectual rights, rules of proper referencing and citing the resources. Different approach in this study year, as stated in section 3, already resulted in decreasing the plagiarism in our programming course.

REFERENCES

Ahtiainen, A., Surakka, S., Rahikainen, M., 2006. Plaggie: GNU-licensed source code plagiarism detection engine for Java exercises. In *Proceedings of the 6th Baltic Sea conference on Computing education research: Koli Calling 2006*. ACM, pp. 141-142.

Austin, M., Brown, L., 1999. Internet plagiarism: Developing strategies to curb student academic dishonesty. *The Internet and higher education*, 2(1), pp. 21–33.

Baggaley, J., Spencer, B., 2005. The mind of a plagiarist. *Learning, Media & Technology*, 30(1), pp. 55-62.

Bennett, R., 2005. Factors associated with student plagiarism in a post-1992 university. *Assessment & Evaluation in Higher Education*, 30(2), pp. 137-162.

Bowyer, K. W., Hall, O. L., 1999. Experience Using ”MOSS” to Detect Cheating On Programming Assignments. In: *Frontiers in Education Conference, FIE '99, 29th Annual, Puerto Rico*. pp. 18-22.

Clough, P., 2000. Plagiarism in natural and programming languages: an overview of current tools and technologies. *Technical Report, Sheffield University*, pp. 1-31.

Culwin, F., MacLeod, A., Lancaster, T., 2001. Source code plagiarism in UK HE computing schools, Issues, attitudes and tools. *Technical Report SBU-CISM-01-01*, Joint Information Committee, School of computing, information systems & mathematics, South Bank University, London, pp. 1-34.

Faidhi, J.A.W., Robinson, S.K., 1987. An empirical approach for detecting similarity and plagiarism within a university programming environment. *Computers and Education*, 11(1), pp. 11-19.

Frantzeskou, G., Macdonell, S., Stamatatos, E., Gritzalis, S., 2008. Examining the significance of high-level programming features in source code author classification. *Journal of Systems and Software*, 81(3), pp. 447-460.

Hammond, M., 2004. Cyber plagiarism: are FE students getting away with words. In *Plagiarism: Prevention, Practice and Policies 2004 Conference, Newcastle*. Northumbria University Press, pp. 257-264.

Joy, M., Luck, M., 1999. Plagiarism in programming assignments. *IEEE Transactions on Education*, 42(2), pp. 129-133.

Moussiades, L., Vakali, A., 2005. PDetect: A Clustering Approach for Detecting Plagiarism in Source Code Datasets. *The Computer Journal*, 48(6), pp. 651-661.

Niggemann, O., 2001. Visual data mining of graph-based data. PhD Thesis, Paderborn University, Paderborn.

Parker, A., Hamblen, J., 1989. Computer algorithms for plagiarism detection. *IEEE Transactions on Education*, 32(2), pp. 94-99.

Prechlet, L., Malpohl, G., Philippsen, M., 2000. JPlag: Finding plagiarisms among a set of programs. Technical Report 2000-1, Fakultät für Informatik, Universität Karlsruhe, Karlsruhe.

Schiller, R.M., 2005. E-Cheating: Electronic Plagiarism. *Journal of the American Dietetic Association*, 105(7), pp. 1058-1062.

Sraka, D., Kaučič, B., 2009. Source Code Plagiarism. In Proceedings of Information Technology Interfaces ITI2009, Cavtat, Croatia. pp. 461-466.

Zeidman, R., 2006. Software Source Code Correlation. In: *5th IEEE/ACIS International Conference on Computer and Information Science, 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse (ICIS-COMSAR'06)*. IEEE Computer Society.