Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

Annales UMCS Informatica AI XI, 1 (2011) 89–101
DOI: 10.2478/v10065-011-0023-6

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI
http://www.annales.umcs.lublin.pl/

Python and C#: a comparative analysis fromsStudents' perspective

Jakub Swacha[1*], Karolina Muszyńska[1†]

[1] *Institute of Informatics Technology in Management, University of Szczecin, Mickiewicza 64,71-101 Szczecin*

Abstract – Python and C# are two programming languages of great educational value. Python has a simple and clear syntax, as well as a concise and readable source code, but is relatively slow, and its industrial applications are mostly web-based. Although the syntax and code readability of C# does not match the Python's high level, they are not very bad either; besides, the .NET language offers a fast JIT compiler, and can be found within a wide gamut of industrial applications. In this paper we present a comparison of these two languages, based on the data acquired from IT students who learnt both of them during programming courses. We use the Analytic Hierarchy Process (AHP) technique to show the dominance of one language over another in respective comparison criteria, and obtain an overall answer which of the two languages is better, according to the ratings given by students.

1 Introduction

Learning computer programming rises a number of difficulties [**1**], which may be greatly diminished if a proper programming language is chosen for a course, depending on its aims and the level of students' programming skills prior to the course [**2**]. Having two languages with different educational advantages, such as Python and C#, the choice becomes difficult. In this paper we aim to show that the useful information for making this choice can be obtained from the students themselves, and how it can be used to obtain a simple answer which of the two languages is better, according to the students. A valuable output of our survey are the questionnaire results that shed some light on how Python and C# programming languages are viewed by students in respective comparison criteria.

*jakubs@uoo.univ.szczecin.pl
†karolina.muszynska@wneiz.pl

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

2 Research methodology

The goal of this research is to determine which of the two programming languages (C# and Python) better suits students' needs from the point of view of the students themselves.

In order to establish the students' opinion on this subject, we used a questionnaire approach, following many examples in the literature on comparing programming languages (see, e.g., **[3, 4, 1, 5]**). An on-line questionnaire has been prepared, containing 16 questions, of which 12 questions were directly related to the survey, and the remaining four questions were intended for better description of the surveyed group.

Every question of the main group corresponds to a single criterion of evaluation, thus their answers give a good glimpse into how the two programming languages are viewed by the students.

In order to obtain an answer to the main research question, we decided to resort to a multicriteria decision analysis method. There are quite many such methods available **[6, 7]**; the most popular of them are **[8]**:

- AHP (Analytic Hierarchy Process) **[9]**,
- ELECTRE (ELimination Et Choix Traduisant la REalité) **[10]**,
- MAUT (Multiattribute Utility Theory) **[11]**,
- PROMETHEE (Preference Ranking Organization METHod for Enrichment Evaluations) **[12]**,
- SMART (Simple Multiattribute Ranking Technique)**[13]**,
- UTA (UTilitès Additives) **[14]**.

Not each of these methods would be equally useful for the analysis of programming language choice. Some of them allow the non-comparability between two solutions (thus, not producing clear and unequivocal results), require to specify many complicated parameters, disregard criteria weights, or are too complex computationally **[8]**.

Following the notable example of A. S. Jadhav and R. M. Sonar and around 20 research works quoted by them **[15]**, we decided to base our methodology on the AHP method. Below is a short description of the used procedure.

Analytic Hierarchy Process (AHP) is a multiple criteria decision making technique, which provides a rational framework for structuring a decision issue, representing and quantifying its elements, for relating these elements to overall goals and for evaluating alternative solutions. TheAHP method is based on the utility function of the decision maker, assumes comparability of criteria and is implemented in four main stages **[9]**:

- creating decision hierarchy,
- defining importance of criteria,
- defining preferences in relation to alternatives,
- calculating the overall measure, which is used to make a decision.

The reason for applying theAHP method is to determine the ranking of alternatives - in this case programming languages, taking into account different evaluation criteria. In the case study presented in this paper selection of the preferable programming language

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

from the students' perspective is based on twelve criteria concerning such aspects as: effectiveness, productivity, stability, ease-of-use (the exact list of criteria is shown in fig.1 in the decision hierarchy).

The top of the hierarchy is the place for the goal of the decision process – in this case it is the selection of the preferred programming language, next levels show criteria, sub-criteria and alternatives - in this case we have 12 criteria and two alternatives – the programming languages: Python and C# (see Fig. 1).

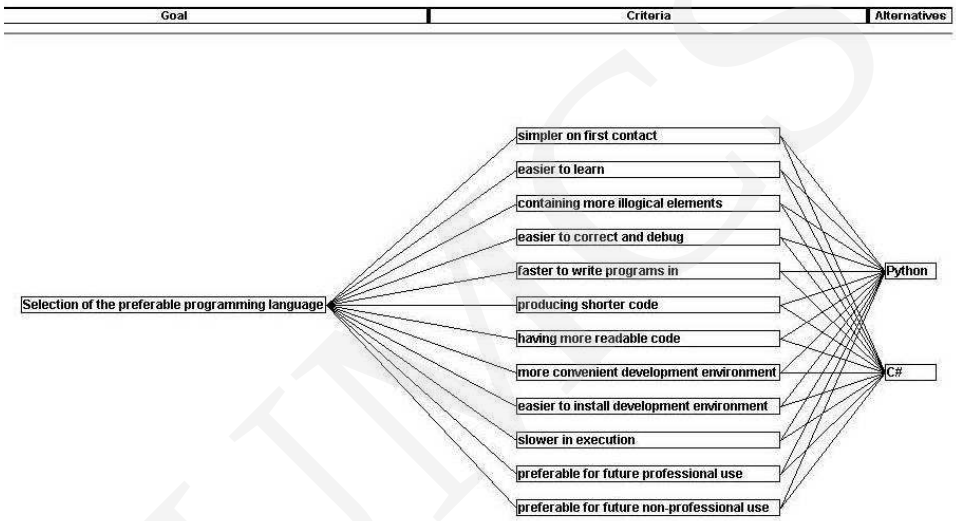

Fig. 1. Decision hierarchy for selecting the preferable programming language.

The criteria were not assumed to be equally important. The authors, basing on their own experience, defined weights for each criterion themselves (see Table 1).

Table 1. Weighting the criteria importance.

Criterion	**Weight**
Simple on the first contact	0.04
Easier to learn	0.08
Less illogical or counter-intuitive	0.10
Easier to find and debug errors	0.10
Allows to write programs faster	0.10
Has shorter source code	0.10
More readable source code	0.10
More convenient development environment	0.06
Easier to install and use at home	0.04
Faster program build and execution	0.08
Useful for professional programmers	0.10
Useful for non-professional programmers	0.10
Sum	1.00

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

In the evaluation phase the alternatives are compared directly by their values, or, as in our case, using a scale introduced by Saaty, in which verbal judgments correspond to numerical values:

- alternatives equal – 1,
- minor dominance – 3,
- strong dominance – 5,
- very strong dominance – 7,
- absolute dominance – 9,
- intermediary values – 2, 4, 6, 8.

The evaluation procedure therefore consisted in comparing both alternatives – the programming languages – in regard to each of the criteria.

3 Respondents

In our survey, the students – respondents of the questionnaire – took the role of experts, and performed the comparisons. Over 200 Information Technology and Econometrics students of the 2nd and 3rd years were asked to fill in the questionnaire published on-line, of which 106 did so. Over 77% of the respondents were men, whereas about 23% were women. The age of the respondents is shown in Fig. 2.

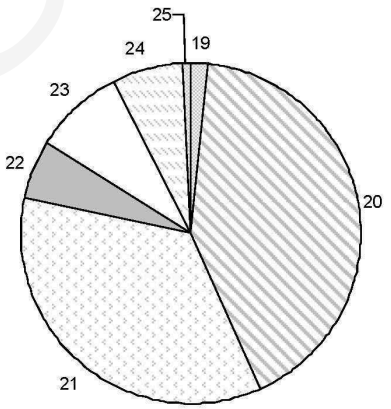

Fig. 2. The age of the respondents.

Over 63% of the students assessed their knowledge of Python as 'average or better', whereas over 86% of the students assessed their knowledge of C# as 'sub-average'. This bias will be taken into consideration when drawing conclusions.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

4 Questionnaire and students' answers

The students were given the following questions relevant to the Python/C# comparison:

1. Which language appeared to you as simpler on the first contact?

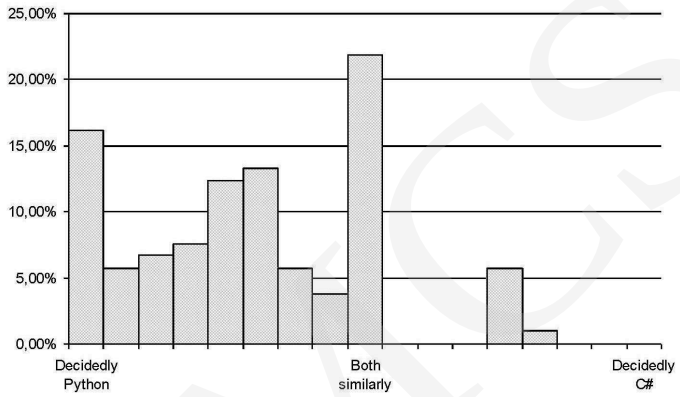

Fig. 3. Answers to question 1.

As Fig. 3 shows, Python is viewed as a simple languager on the first contact by the majority of the students.

2. Learning which language was easier for you?

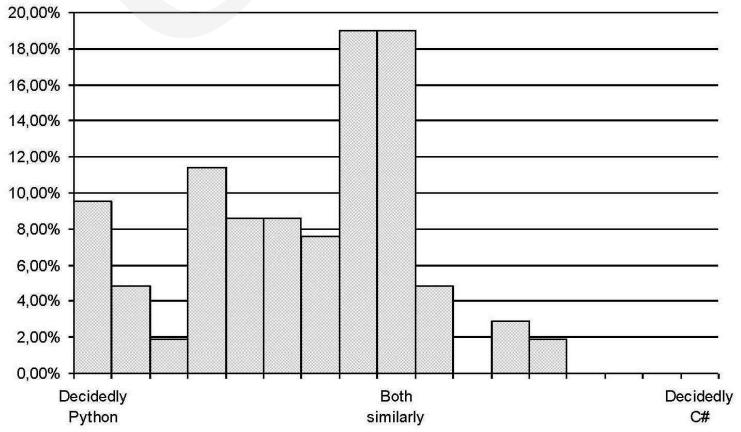

Fig. 4. Answers to question 2.

More than half of the students assessed both languages as equally or nearly equally difficult to learn (see Fig. 4). No one picked C# as decidedly easier.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

3. Which language contains more syntax elements which are illogical or counter-intuitive, according to you?

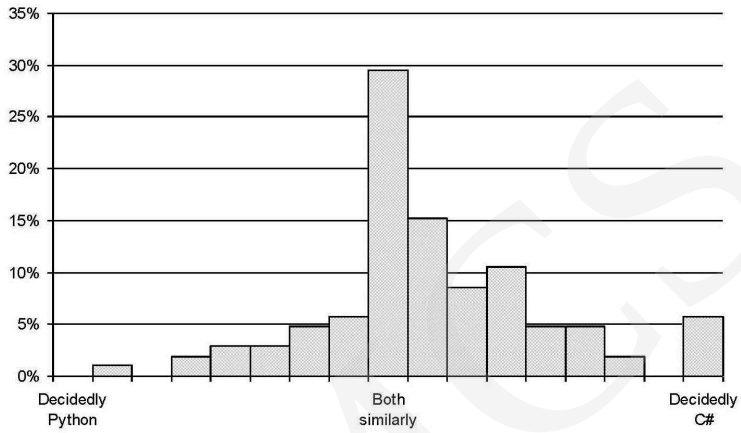

Fig. 5. Answers to question 3.

Most students found equally many illogical and/or counter-intuitive elements in both languages (see Fig. 5). There were more decided critics of C# than of Python.

4. In a program in which language you find and debug errors easier?

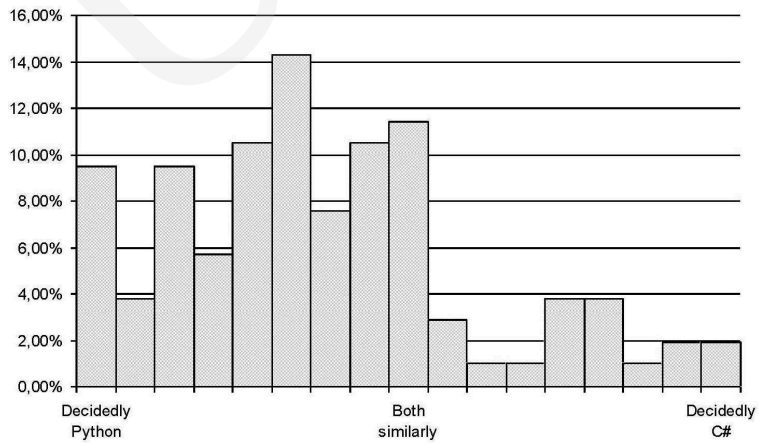

Fig. 6. Answers to question 4.

Significantly more students feel it easier to debug Python programs than those in C# (see Fig. 6). It is surely connected to their level of language command (lower in the case of C# than of Python), much more than previous answers.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

5. If a programmer knows both the languages equally well, in which of them will he write an average program faster?

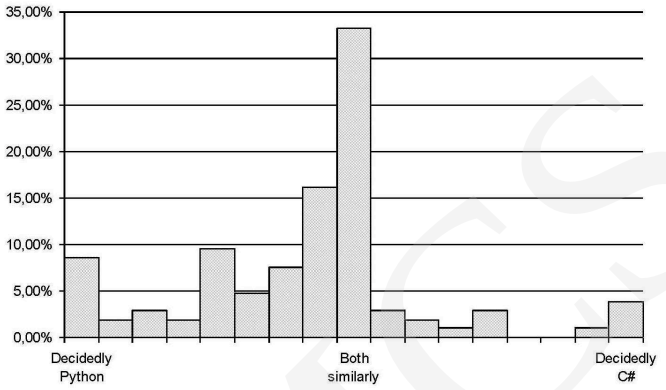

Fig. 7. Answers to question 5.

In this question we asked the students to assume that the level of language command of both languages is equal. With this assumption, most students expect the programmer to write programs equally fast in C# and Python (see Fig. 7). More students pick Python as decidedly faster to write programs than C#.

6. If two programs do the same thing, which of them will have the shorter source code?

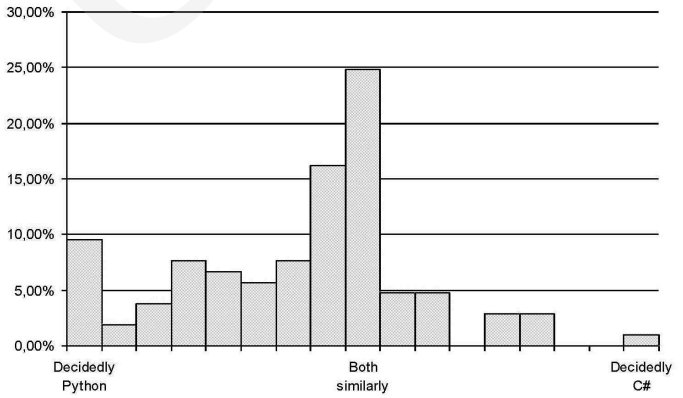

Fig. 8. Answers to question 6.

The answers to this question (see Fig. 8) are similar to the previous one. Although most of the students' see both languages nearly equally fast to write programs, more students pick Python as decidedly faster to write programs than C#.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

7. If two programs do the same thing, which of them will have the more readable source code?

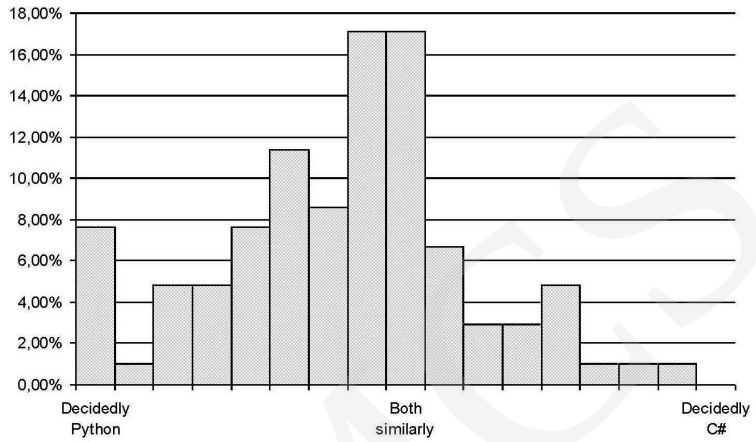

Fig. 9. Answers to question 7.

Again, most of the students' see both languages' source code nearly equally readable (see Fig. 9) with a bigger group pointing to the source code in Python as decidedly more readable than that in C#.

8. Which language has more convenient software development environment?

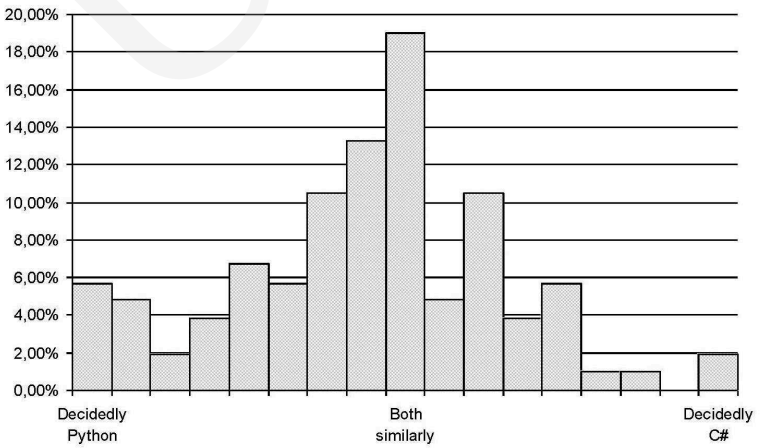

Fig. 10. Answers to question 8.

Most of the answers to question 8 (see Fig. 10) show the two languages to be nearly equal, like the previous one. Once again we can see Python's advantage among 'decidedly' answers.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

9. Which language's software development environment is easier to install and use at home?

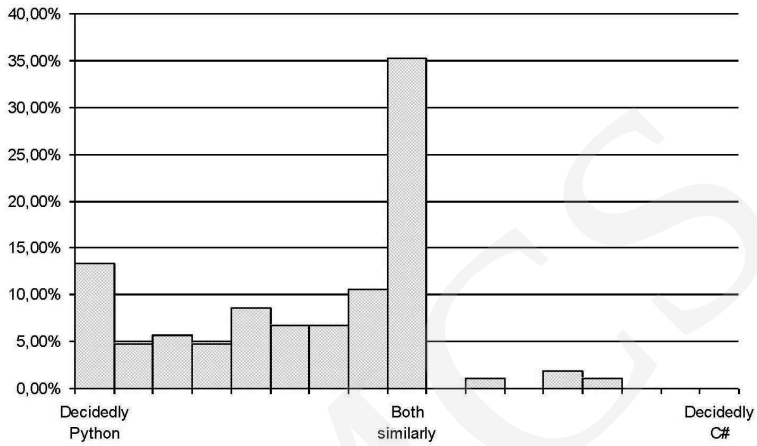

Fig. 11. Answers to question 9.

Although the group of students who see both languages equal is once again the most numerous one, the answers to this question clearly point to Python as the language with software development platform which is easier to install and use at home (see Fig. 11).

10. In the case of which language, the slowness (of both the programs and the development environment) is more of an issue to you?

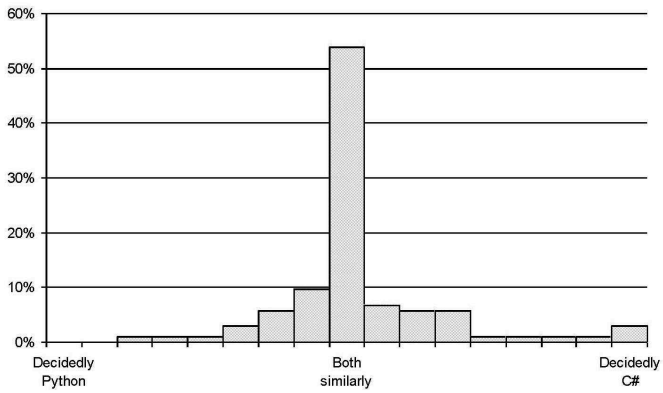

Fig. 12. Answers to question 10.

It seems that (see Fig. 12), to the students, the slowness of code execution in Python is as much an issue as the slowness of program building in C#.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

11. If you work in the future as a programmer, which of the two languages you would rather use?

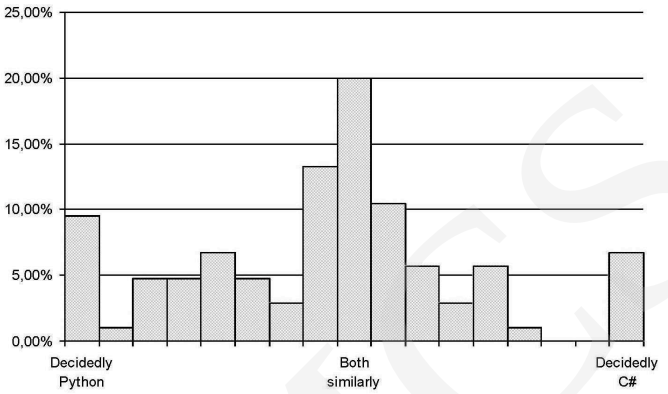

Fig. 13. Answers to question 11.

Even though the students are supposed to be well aware of the current domination of C-derived languages in both world-wide and local programming industry, they see both languages as almost equally useful for them as professional programmers of future (see Fig. 13).

12. If you work in the future in a position other than a programmer, which of the two languages you would rather use?

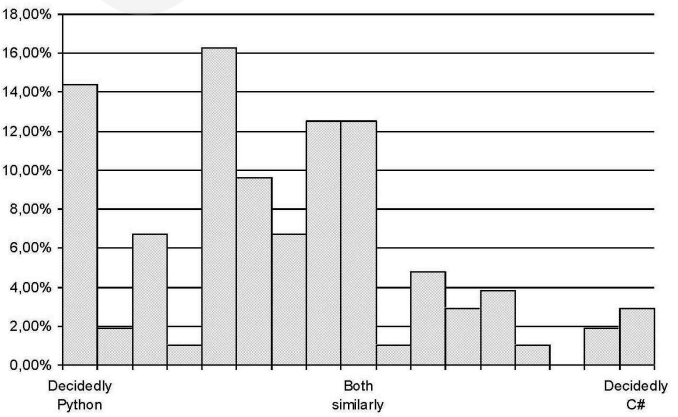

Fig. 14. Answers to question 12.

When deciding about their language of choice for non-professional programming, most of them point to Python (see Fig. 14). Of course, this should be

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

connected to their current level of knowledge of both languages, still it should not be overemphasized.

5 Analysis of questionnaire results

In the considered case we have a numerous group of experts (over one hundred) who gave their preferences concerning two alternative programming languages within each criterion. There are three basic approaches that a group can use to assess weights: consensus, vote or compromise and geometric or arithmetic means of the individual judgments. In the first approach, the group of decision makers is required to reach a consensus on each judgment. If the group is unable to reach a consensus, then a vote or compromise is used in the second approach to set the judgments values. In the third approach, let a_{ij}^k denote the comparison of element i to element j for a decision maker k $(k = 1, 2, \ldots, n)$ in the pairwise comparison matrix. The individual judgments of the n decision makers are combined using the geometric means to produce the entry $a_{ij} = [a_{ij}^1 \cdot a_{ij}^2 \cdot a \ldots \cdot a_{ij}^n]^{1/n}$. Aczél and Saaty [**16**] have shown that the geometric means is the only one that preserves the reciprocal property in the combined pairwise comparison matrix, for which reason it is the most common approach used to set priorities [**17**].

Based on this approach, geometric means within each criterion has been calculated, and the results used in the evaluation phase. For defining a decision hierarchy (Fig. 1), criteria importance and preferences in relation to alternatives, Criterium DecisionPlus application was used. In Fig. 15 the overall score for decision of selecting the preferable programming language is presented. The graph shows the overall dominance of Python over C# (0.618 versus 0.382), which indicates that it is the preferred language from the students' point of view.

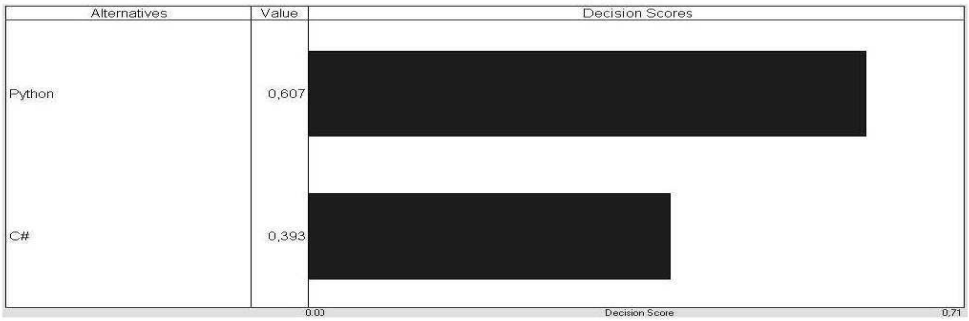

Fig. 15. Decision Scores – results of the decision model.

In order to see which of the criteria mostly influenced the decision score contribution by criteria view can be selected. The areas of the bars in Fig. 16 correspond to the criteria listed to the right of the graph, representing the amount of contribution each criterion made to that alternative decision score.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

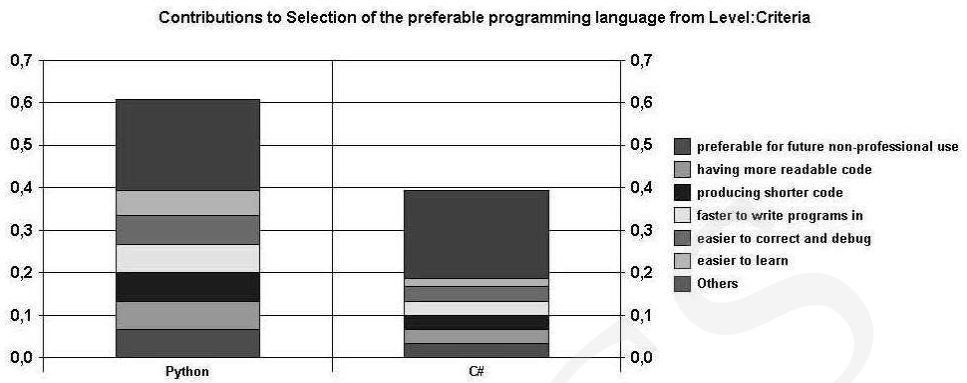

Fig. 16. Contributions by criteria – results of the decision modell.

6 Conclusions

The results of our survey clearly point to Python as a better language than C#, according to the students who responded to the questionnaire. Of the 12 questions asked, Python completely dominated C# in the case of five of them, attaining clearly visible prevalence in the case of the other three; for the remaining four questions, the answers were close to even. The combined result, obtained using the AHP method, points to Python as a better language in overall, still the dominance over C# (0.618 to 0.382) is not as high as the detailed results would suggest.

The result should be interpreted in the context of the fact that most of the students rated their knowledge of Python higher than that of C#. Still, some of the answers to the detailed questions indicate that the rate could have been higher in favor of C#, if it was free of the shortcomings that make it more difficult to learn.

References

[1] Lahtinen E., Ala-Mutka K., Järvinen H.-M., A study of the difficulties of novice programmers, [in:] Proceedings of the 10th annual SIGCSE conference on Innovation and technology in computer science education, ACM, New York (2005): 14.

[2] Swacha J., New concepts for teaching computer programming to future Information Technology engineers, [in:] Perspective Technologies and Methods in MEMS Design, Lviv Polytechnic National University, Lviv (2010): 188.

[3] Bowyer P., An investigation into teaching introductory programming to physics undergraduates, University of Southampton, Southampton (2006).

[4] Chen X., Kurtonina N., Taylor S., First Programming Languages Revisited, [in:] Proceedings of the 2004 College Teaching & Learning Conference, Lake Buena Vista (2004).

[5] Madden M., Chambers D., Evaluation of student attitudes to learning the Java language, [in:] Proceedings of the inaugural conference on the Principles and Practice of programming, National University of Ireland, Maynooth (2002): 125.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 28/11/2026 12:18:45

[6] Figueira J., Greco S., Ehrgott M. (eds.), Multiple criteria decision analysis: state of the art surveys, Springer, New York (2005).
[7] Olson D. L., Decision aids for selection problems, Springer-Verlag, New York (1996).
[8] Ziemba P., Piwowarski M., Metody analizy wielokryterialnej we wspomaganiu porównywania produktów w Internecie, Metody Informatyki Stosowanej 2 (2008): 137.
[9] Saaty T. L., The Analytic Hierarchy Process, McGraw-Hill, New York (1980).
[10] Roy B., The outranking approach and the foundations of Electre methods, Theory and decision 31 (1991): 49.
[11] Dyer J. S., Fishburn P. C., Steuer R. E., Wallenius J., Zionts S., Multiple Criteria Decision Making, Multiattribute Utility Theory: The Next Ten Years, "Management Science" 38(5) (1992): 645.
[12] Brans J.P., Vincke P., A preference ranking organisation method: The PROMETHEE method for MCDM, Management Science 31(6) (1985): 647.
[13] Edwards W., Barron F. H., SMARTS and SMARTER: Improved simple methods for multiattribute utility measurement, Organizational Behavior and Human Decision Processes 60 (1994): 306.
[14] Beuthe M., Scannella G., Comparative analysis of UTA multicriteria methods, European Journal of Operational Research 130(2) (2001): 246.
[15] Jadhav A. S., Sonar R. M., Evaluating and selecting software packages: A review, Information and Software Technology 51(3) (2009): 555.
[16] Aczél J., Saaty T. L., Procedures for synthesizing ratio judgements, Journal to of Mathematical Psychology 27(1) (1983): 93.
[17] Melón M. G., Beltran P. A., Cruz M. C. G., An AHP-based evaluation procedure for Innovative Educational Projects: A face-to-face vs. computer-mediated case study, Omega – The International Journal of Management Science 36 (2008): 754.