UDC 004.9: 004.05
DOI: 10.31891/CSIT-2021-4-1

VASYL STETSYUK, TETIANA HOVORUSHCHENKO
Khmelnytskyi National University

METHOD OF CHOOSING THE PROGRAMMING ENVIRONMENT FOR SOFTWARE

This paper shows an example of the application of the method of hierarchy analysis to build a hierarchy of programming environments, which provides support for selecting the optimal programming environment for software in accordance with the requirements of the developer and user. As a result of the application of the method of hierarchy analysis, a hierarchy of programming environments for software development was built, which will be useful for building criteria and production rules for selecting a programming environment for software. The constructed hierarchy has the following form: 1) Microsoft Visual Studio (33.1%); 2) Eclipse (19.6%); 3) PhpStorm (19.2%); 4) Netbeans (17.2%); 5) PyCharm (11%). Hierarchy analysis makes it possible to determine what is the best for software development is the Microsoft Visual Studio environment, but the price of such an environment is quite high. Next and almost equal in technical capabilities are the environments Eclipse, PhpStorm, Netbeans, PyCharm.

Keywords: software, programming environment, method of hierarchy analysis.

ВАСИЛЬ СТЕЦЮК, ТЕТЯНА ГОВОРУЩЕНКО
Хмельницький національний університет

МЕТОД ВИБОРУ СЕРЕДОВИЩА ПРОГРАМУВАННЯ ДЛЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Наразі перед будь-якою компанією, що займається розробленням програмного забезпечення (ПЗ), постає задача вибору середовища програмування. Очевидно, що цей вибір зводиться до багатокритеріальної задачі і далеко не очевидний. Багатокритеріальність вибору середовища програмування полягає у тому, що кожне існуюче середовище програмування слід оцінювати не за одним критерієм, а за сукупністю багатьох показників (критеріїв), що розглядаються одночасно.

У статті дістала подальшого розвитку математична модель середовища програмування, яка дає можливість оцінити середовище програмування для ПЗ більш точно, з врахуванням всіх його складових частин, а також дає можливість побудувати критерії та продукційні правила вибору середовища програмування для ПЗ, які підтримуватимуть організацію у виборі середовища програмування для програмного забезпечення.

У даній статті показано приклад застосування методу аналізу ієрархій для побудови ієрархії середовищ програмування, яка надає підтримку вибору оптимального середовища програмування для системного програмного забезпечення відповідно до вимог розробника та користувача.

В результаті застосування методу аналізу ієрархій було побудовано ієрархію середовищ програмування для розроблення ПЗ, яка буде корисною для побудови критеріїв та продукційних правил вибору середовища програмування для ПЗ. Побудована ієрархія має наступний вигляд: 1) Microsoft Visual Studio (33,1%); 2) Eclipse (19,6%); 3) PhpStorm (19,2%); 4) Netbeans (17,2%); 5) PyCharm (11%). Аналіз ієрархії дає можливість визначити, що найкращим для ПЗ є середовище Microsoft Visual Studio, але ціна такого середовища є доволі високою. Наступними і практично рівними за технічними можливостями є середовища Eclipse, PhpStorm, Netbeans, PyCharm.

Перспективним напрямком подальших досліджень є проектування та реалізація системи підтримки прийняття рішень для вибору середовища програмування ПЗ на основі запропонованої концепції використання методу аналізу ієрархії.

Ключові слова: програмне забезпечення, середовище програмування, метод аналізу ієрархії.

Introduction

At present, any software development company is faced with the task of choosing a programming environment. Obviously, this choice is reduced to a multi-criteria task and is far from obvious. The multi-criteria choice of the programming environment is that each existing programming environment should be evaluated not by one criterion, but by a set of many indicators (criteria) considered simultaneously.

The developer can find a lot of information about the programming environment on the Internet, but this information is not structured. Software developers spend a lot of time finding the best solution for choosing a programming environment, reading forums and seeking advice from more experienced developers, but such advice may not be objective enough, because, as you know, choosing a programming environment is mostly a matter of habit. Software developers typically use programming environments that they have been working with for a long time and are accustomed to the features (advantages and disadvantages). Mastering a new programming environment requires a lot of time and money, and the price of a modern programming environment for software development can reach $ 1,000 per license. It is clear that if you need to provide a staff of programmers, say, a small company of 10-30 employees, by programming environment, the costs become quite significant. The important question is whether such a high price is realistic for existing programming environments. But much more important is the question of whether a high price is real for the programming environment in the context of the task. It is possible that a cheaper or free environment will be enough to perform the tasks set before the programmer. Thus, the current task is to support the choice of a programming environment for software, which is to choose the best solution from a variety of possible (optimization) and order possible solutions by priority (ranking)

Given the above, the purpose of this study is to create a mathematical model of the programming environment, as well as to support the selection of the optimal programming environment in accordance with the requirements of the developer and user.

Mathematical model of the programming environment

Five well-known programming environments were chosen as elements of choice (alternatives): Microsoft Visual Studio, PhpStorm, PyCharm, Eclipse, Netbeans [1].

The programming environment is used for software development and is focused on a specific language or several programming languages [2]. In general, the main characteristics of the programming environment are: available programming languages, operating systems with which the developed programs are compatible, price, interface, availability of components for testing.

Based on the analysis of modern programming environments and the characteristics of the programming environments, let's develop a mathematical model of the programming environment:

$$PE = \{APL, OSC, Cost, Int, ACT\}, \qquad (1)$$

where $APL = \{apl_1,...,apl_n\}$ – set of programming languages available in the environment (n – number of available programming languages); $OSC = \{osc_1,...,osc_m\}$ – the set of operating systems with which the applications developed in the environment are compatible (m – the number of operating systems with which the developed applications are compatible); $Cost$ – the price level of the programming environment (for example, on a 5-point scale (a score of "5" indicates the lowest price level)); Int – expert evaluation of the interface (for example, on a 5-point scale (score "5" indicates the best interface)); $ACT = \{act_1,...,act_k\}$ – the set of components of the environment intended for testing (k – the number of components for testing).

Method of choosing the programming environment for software based on the method of hierarchy analysis

To apply the problem of choosing the optimal programming environment, we use the method of hierarchy analysis (MHA) [3-6], the main stages of which are: 1) decomposition of the problem into components that are easy to compare; 2) distribution of components by levels and establishment of essential connections; 3) pairwise comparison of elements of each level; 4) synthesis of all levels and ordering of elements of choice by priority for the decision-maker.

We will decompose the problem into components based on the method of hierarchy analysis [3, 7]:

Level 0 – selection of software programming environment.

Level 1 – hierarchy forces: for FreeWare (free software); for Commercial Software (paid software).

Level 2 – actors (participants) of the hierarchy: junior programmer (Junior); programmer; tester.

Level 3 – goals of the actors: Handbook; Ease of use; Code readability; Productivity; Writing speed; The speed of the program work; The final size of the distribution; Number of programming languages; Speed of development, number of components; Debugger.

Level 4 – alternatives: Microsoft Visual Studio, PhpStorm, PyCharm, Eclipse, Netbeans.

The results of the comparison of the elements of the 4th level in relation to the criteria of the 3rd level are summarized in Table 1, in which the alternatives are ranked by experts starting from one (1) – for the best element to four (4) – for the worst element.

Table 1

Description of alternatives

Alternatives Criteria	Microsoft Visual Studio	PhpStorm	PyCharm	Eclipse	Netbeans
Directory	1	1	3	2	4
Ease of use	2	2	4	4	1
Code readability	1	2	3	3	1
Productivity	1	1	3	2	4
Writing speed	2	3	4	4	1
The speed of the program	1	1	3	3	4
The final size of the program	1	1	2	1	3
Number of programming languages	1	4	3	2	4
Speed of development	1	2	2	2	3
Number of components	3	2	4	2	4
Debugger	1	1	3	2	2

As can be seen from Table 1, the highest rank for most of the selected criteria was assigned to Microsoft Visual Studio, which previously indicates the superiority of this environment. It can also be seen that alternative 3 - PyCharm - does not outperform the others by any criteria.

It should also be noted that this hierarchy did not consider such an important and weighty selection criterion as price, as the participants in the hierarchy were those who use the software environment as a tool in the work, rather than solving commercial issues. As can be seen from the description of alternatives, they all differ in price and when included in the hierarchy of the participant responsible for the commercial side of the issue, such as CFO or CEO, whose weight in the final choice is usually very significant, the final qualitative and quantitative ordering of elements may be fundamentally different [3, 7].

According to the number of elements of the hierarchy of each level and the established links, the decision-makers together with experts fulfilled the matrices of pairwise comparisons according to the Saati scale, checked the consistency of the constructed matrices by means of the maximum eigenvalue of the matrices of pairwise comparisons $\lambda_{\max}$, the ratio of consistency, random consistency index, coherence ratio, found the vectors of local priorities p using the main eigenvector v [3-7].

For example, at the 0th level, the matrix of pairwise comparisons is filled, according to which the forces of the hierarchy are organized – free and paid software environment – according to their contribution to the achievement of the main goal (Table 2).

Table 2

Matrix of pairwise comparisons of the 0th level

Choosing a programming environment	Free	Paid
Free	1	1/5
Paid	5	1

The last step in the decomposition of the problem is the formation of matrices of local priorities of each level from the corresponding vectors of local priorities A_i, $i = \overline{0, m-1}$, where m – the number of levels [3-7] – Tables 3-6. Tables 4-7 highlight the best selection elements according to the relevant criteria..

Table 3

Matrix of local priorities of the 0th level – A_0

	Vectors of local priorities
Free	0,167
Paid	0,833

Table 4

Matrix of local priorities of the 1st level – A_1

	Free	Paid
Junior programmer (Junior)	0,122	0,066
Programmer	0,558	0,382
Tester	0,320	0,551

Table 5

Matrix of local priorities of the 2nd level – A_2

	Junior programmer (Junior)	Programmer	Tester
Directory	0,252		
Ease of use	0,161		
Code readability			0,188
Productivity		0,358	
Writing speed	0,101	0,103	
The speed of the program		0,158	
The final size of the program		0,03	0,081
Number of programming languages	0,065	0,067	
Speed of development	0,376		
Number of components	0,045	0,239	
Debugger		0,045	0,731

The second main stage of the decision support process – synthesis – is carried out from the lower level to the upper, while consistently arranging alternatives to all criteria of previous levels, in particular, in relation to the main goal of the hierarchy [3-7] – Table 7-9.

As a result of the application of the method of hierarchy analysis, a hierarchy of programming environments for software development was built, which will be useful for building criteria and production rules for selecting a programming environment for software. The constructed hierarchy has the following form: 1) Microsoft Visual Studio (33.1%); 2) Eclipse (19.6%); 3) PhpStorm (19.2%); 4) Netbeans (17.2%); 5) PyCharm (11%). Hierarchy analysis makes it possible to determine what is the best for software development is the Microsoft Visual Studio environment, but the price of such an environment is quite high. Next and almost equal in technical capabilities are the environments Eclipse, PhpStorm, Netbeans, PyCharm.

Table 6

Matrix of local priorities of the 3rd level – A_3

	Directory	Ease of use	Code readability	Productivity	Writing speed	The speed of the program	Number of programming languages	The final size of the program	Speed of development	Number of components	Debugger
MS Visual Studio	0,387	0,179	0,313	0,376	0,257	0,323	0,260	0,333	0,298	0,169	0,369
PhpStorm	0,212	0,149	0,176	0,215	0,153	0,292	0,260	0,288	0,298	0,425	0,109
PyCharm	0,119	0,081	0,099	0,121	0,088	0,154	0,138	0,152	0,158	0,068	0,109
Eclipse	0,212	0,081	0,099	0,215	0,088	0,154	0,260	0,152	0,158	0,270	0,206
Netbeans	0,069	0,510	0,313	0,074	0,415	0,078	0,082	0,075	0,089	0,068	0,206

Table 7

The choice of programming environment in relation to the actors and their goals – $A_{3,2} = A_3 \cdot A_2$

	Junior programmer (Junior)	**Programmer**	**Tester**
MS Visual Studio	0,293	0,299	0,350
PhpStorm	0,243	0,272	0,134
PyCharm	0,124	0,112	0,110
Eclipse	0,158	0,202	0,190
Netbeans	0,183	0,115	0,216

Table 8

The choice of programming environment in relation to the forces of the hierarchyn– $A_{3,2,1} = A_{3,2} \cdot A_1$

	Free	**Paid**
MS Visual Studio	0,315	0,327
PhpStorm	0,225	0,194
PyCharm	0,113	0,112
Eclipse	0,193	0,192
Netbeans	0,155	0,175

Table 9

The choice of programming environment in relation to the main purpose of the problem –

$$A_{3,2,1,0} = A_{3,2,1} \cdot A_0$$

	Choosing a programming environment	**Rank**
MS Visual Studio	0,331	1
PhpStorm	0,192	3
PyCharm	0,11	5
Eclipse	0,196	2
Netbeans	0,172	4

Conclusions

At present, any software development company is faced with the task of choosing a programming environment. Obviously, this choice is reduced to a multi-criteria task and is far from obvious. The multi-criteria choice of the programming environment is that each existing programming environment should be evaluated not by one criterion, but by a set of many indicators (criteria) considered simultaneously.

The current task is to support the choice of a programming environment for software, which is to choose the best solution from a variety of possible (optimization) and order possible solutions by priority (ranking). The purpose of this study is to create a mathematical model of the programming environment, as well as to support the selection of the optimal programming environment in accordance with the requirements of the developer and user.

The paper further develops the mathematical model of the programming environment, which allows evaluating the programming environment for software more accurately, taking into account all its components, and also allows to build criteria and production rules for choosing a programming environment for software that will support the organization in choosing a programming environment for software.

This paper shows an example of the application of the method of hierarchy analysis to build a hierarchy of programming environments, which provides support for selecting the optimal programming environment for software in accordance with the requirements of the developer and user.

As a result of the application of the method of hierarchy analysis, a hierarchy of programming environments for software development was built, which will be useful for building criteria and production rules for selecting a programming environment for software. The constructed hierarchy has the following form: 1) Microsoft Visual Studio (33.1%); 2) Eclipse (19.6%); 3) PhpStorm (19.2%); 4) Netbeans (17.2%); 5) PyCharm (11%). Hierarchy analysis makes it possible to determine what is the best for software development is the Microsoft Visual Studio environment, but the price of such an environment is quite high. Next and almost equal in technical capabilities are the environments Eclipse, PhpStorm, Netbeans, PyCharm..

A promising area of further research is the design and implementation of a decision support system for the selection of software programming environment based on the proposed concept of using the method of hierarchy analysis.

REFERENCES

1. Top 10 IDEs // [Electronic resource] – Access mode: https://timeweb.com/ru/community/articles/5-luchshih-ide-1

2. Development environments in programming // [Electronic resource] – Access mode: https://it-black.ru/sredy-razrabotki-v-programmirovanii/

3. Pichugina O. Application of decision-making methods when choosing a programming environment by an IT company / O. Pichugina // Popular science journal in the field of information, computer and telecommunication technologies, processes and tools "Information infrastructures and technologies". – 2010. – №1. – Pp.24-28.

4. Demin G. Methods of making managerial decisions: tutorial / G. Demin. – Perm: Perm State National Research University, 2019. – 88 p.

5. Khalin V. Decision theory in 2 volumes. Volume 1: textbook and workshop for bachelor's and master's degrees / V. Khalin [and others.] ; edited by V. Khalin. – Moscow : Yurayt Publishing House, 2019. – 250 p.

6. Khalin V. Decision theory in 2 volumes. Volume 2: textbook and workshop for bachelor's and master's degrees / V. Khalin [and others.] ; edited by V. Khalin. – Moscow : Yurayt Publishing House, 2019. – 350 p.

7. Hovorushchenko T., Malyarchuk R., Lavrinchuk V. Support for selecting a programming environment for system software / T. Hovorushchenko, R. Malyarchuk, V. Lavrinchuk // Proceedings of III All-Ukrainian scientific-practical conference of young scientists and students "Intelligent technologies in system programming". – Khmelnytskyi: PP Gonta, 2014 – pp. 372-381.

Vasyl Stetsyuk **Василь Стецюк**	Lecturer of Computer Engineering & System Programming Department, Khmelnytskyi National University, Khmelnytskyi, Ukraine, e-mail: swmuau@gmail.com. orcid.org/0000-0001-9880-2666, Scopus Author ID: 57218242626	старший викладач кафедри комп'ютерної інженерії та системного програмування, Хмельницький національний університет, Хмельницький, Україна.
Tetiana Hovorushchenko **Тетяна Говорущенко**	DrSc, Professor, Head of Computer Engineering & System Programming Department, Khmelnytskyi National University, Khmelnytskyi, Ukraine, e-mail: govorushchenko@gmail.com. orcid.org/0000-0002-7942-1857, Scopus Author ID: 54420153900 https://scholar.google.com.ua/citations?user=aIJyKc8AAAAJ&hl=uk	доктор технічних наук, професор, завідувач кафедри комп'ютерної інженерії та системного програмування, Хмельницький національний університет, Хмельницький, Україна.