

Г.М. Губаль

Луцький національний технічний університет

**ВДОСКОНАЛЕННЯ СТВОРЕННЯ ТАБЛИЦЬ І ДЕЯКИХ ОБЧИСЛЕНЬ НА МОВІ ПРОГРАМУВАННЯ LATEX ДЛЯ МАТЕМАТИЧНИХ ДОСЛІДЖЕНЬ**

*У статті розглядається система LaTeX. Виконано автоматизацію створення даних у математичних дослідженнях за допомогою створених нових макросів у системі LaTeX на мові програмування LaTeX. Створено нові макроси, які генерують матриці випадкових чисел, розмірності яких можна ввести на етапі виконання макросів, які обчислюють і друкують суми введеної кількості доданків, піднесених до заданого степеня, які друкують рядки введених таблиць при заданій умові, які обчислюють і друкують суму елементів рядків і стовпців введених таблиць.*

*Ключові слова:* LaTeX, макроси, матриця, лічильник, умова.

Н.М. Hubal

**IMPROVEMENT OF CREATION OF TABLES AND SOME CALCULATIONS IN THE LATEX PROGRAMMING LANGUAGE FOR MATHEMATICAL RESEARCH**

*The LaTeX system is considered in this article. The automation of data creation in mathematical research is performed using created new macros in the LaTeX system in the LaTeX programming language. New macros are created that generate matrices of random numbers, the dimensions of which can be entered at the stage of execution of macros that calculate and print the sums of the entered number of terms raised to a given power, that print the rows of the entered tables under a given condition, that calculate and print the sum of the elements of the rows and columns of entered tables.*

*Keywords:* LaTeX, macros, matrix, counter, condition.

**Постановка проблеми.** Система LaTeX широко використовується для написання наукових статей, навчальних посібників, підручників, монографій, що мають складне форматування [1]-[14]. Науковці і дослідники надають перевагу системі LaTeX, оскільки в системі LaTeX можна створювати складні формули, посилання і виноска [15]-[24]. Мова програмування LaTeX дає можливість автоматизувати і спростити роботу із системою за допомогою макросів [13], [14].

Використання макросів дозволяє ефективно керувати елементами, що використовуються багато разів. Макроси покращують читання і оновлення програмного коду та зменшують його дублювання, що підвищує швидкість компіляції і обробки документа LaTeX. Макроси полегшують колективну роботу, оскільки розробники проекту можуть ефективно використовувати та змінювати стандартизовані конструкції.

Створимо нові макроси у системі LaTeX на мові програмування LaTeX, які

– генерують матриці випадкових чисел, розмірності яких можна ввести на етапі виконання макросів;

– обчислюють і друкують суми введеної кількості доданків, піднесених до заданого степеня;

– друкують рядки введених таблиць при заданій умові;

– обчислюють і друкують суму елементів рядків і стовпців введених таблиць.

Ці макроси автоматизують і спрощують створення наукових і навчально-методичних праць, здійснюють обчислення, допомагаючи науковцям у їхніх дослідженнях.

**Аналіз останніх досліджень і публікацій.** При створенні документа LaTeX програмний код компілюється. При цьому команди транслюються в інші коди, які в свою чергу транслюються в інші коди і т.д. [5], [8], [11], [16]-[18], [21], [22]. У системі LaTeX можна генерувати тестові питання для перевірки знань студентів [22] та розробляти свої типи тестових питань [13]. Мова програмування LaTeX дає можливість створювати макроси [14].

**Виклад основного матеріалу.** Створимо матрицю випадкових чисел, розмірність якої можна ввести на етапі виконання макросів. У LaTeX документі `article` підключимо пакети `amsmath`, `forloop` та `pgf`. Створимо нові макроси:

– `\rnd`, що генерує випадкове ціле число від -30 до 30 за допомогою пакета `pgf`;

– після оголошення лічильників `row` і `col`, створимо `\randmatrix`, що генерує матрицю за допомогою оточення `\pmatrix` з пакета `amsmath` з заданим числом рядків і стовпців, які можна вводити з клавіатури. `\randmatrix` використовує два аргументи – число рядків і стовпців. Вкладені цикли `forloop` використані для створення рядків і стовпців матриці за допомогою команди `\rnd`.

Наведемо програмний код для створення матриці випадкових чисел, розмірність якої можна ввести (рис. 1).

```

\documentclass{article}

\usepackage{amsmath}
\usepackage{forloop}
\usepackage{pgf}

\newcommand{\rnd}{\pgfmathparse{int(random(-30,30))}
\pgfmathresult}%

\newcounter{row}
\newcounter{col}

\newcommand{\randmatrix}[2]{%
\begin{pmatrix}
\forloop{row}{1}{\value{row} < #1}{%
\forloop{col}{1}{\value{col} < #2}{%
\rnd &
}%%
\rnd
\\
}%
\forloop{col}{1}{\value{col} < #2}{%
\rnd &
}%%
\rnd
\end{pmatrix}
}%

```

Рис. 1. Створення матриці випадкових чисел

У тілі документа виклинемо створену команду `\randmatrix` і введемо число рядків, наприклад, 10 та число стовпців, наприклад, 7 (рис. 2).

```

\begin{document}

Matrix of inputted size with random elements has the
form


$$\randmatrix{10}{7}$$


\end{document}

```

Рис. 2. Звертання до створеної команди у тілі документа

Наведений програмний код (рис. 1 і рис. 2) генерує результат, представлений на рис. 3

Matrix of inputted size with random elements has the form

$$\begin{pmatrix} -14 & -14 & -1 & 21 & -12 & 26 & 20 \\ -7 & 1 & -8 & -23 & 10 & -28 & 3 \\ -10 & -29 & -2 & -20 & -10 & -11 & 14 \\ 27 & 10 & -13 & 10 & 6 & -15 & -6 \\ 0 & -3 & -13 & -6 & -30 & -7 & 23 \\ 24 & -19 & 30 & -25 & 22 & -30 & 12 \\ -3 & 27 & -17 & 23 & -11 & 17 & 30 \\ -13 & -4 & -25 & 11 & -20 & -22 & -9 \\ 9 & -21 & 7 & -17 & -25 & 3 & 20 \\ 10 & -9 & 24 & 23 & -9 & -16 & 28 \end{pmatrix}$$

Рис. 3. Результат, згенерований програмним кодом

У випадках, коли у наукових дослідженнях треба обчислювати досить складні границі при  $n \rightarrow \infty$ , можна їх обчислювати на комп'ютері для великих  $n$ . При цьому чим більше  $n$ , тим довше виконуються такі обчислення, а при  $n \rightarrow \infty$  обчислення границь на комп'ютері немає змісту.

Можна спростити обчислення складних границь послідовностей при  $n \rightarrow \infty$  за допомогою коду на мові програмування LaTeX. Наприклад, знайдемо границю послідовності з загальним членом

$$a_n = 1^a + 2^a + 3^a + \dots + n^a + \left(\frac{1}{u}\right)^{2n}$$

при  $a=3$ ,  $u=120$ ,  $n \rightarrow \infty$ . Для цього запишемо границю

$$\lim_{n \rightarrow \infty} a_n = 1^a + 2^a + 3^a + \dots + u^a.$$

У цьому випадку можна обчислити границю послідовності за допомогою програмного коду, написаного на мові LaTeX (рис. 4). У LaTeX документі `\article` підключимо пакети `\calc` та `\forloop`. Створимо нові лічильники `\r`, `\t` і `\result`. Присвоїмо значення 0 лічильнику `\result`. Створимо команду `\sumn` для обчислення і друку суми введеної кількості натуральних чисел, піднесених до кубу. Введена кількість натуральних чисел є аргументом цієї команди, і ця кількість повинна бути введена більшою на одиницю.

```
\documentclass{article}
\usepackage{calc}
\usepackage{forloop}

\newcounter{r}
\newcounter{t}
\newcounter{result}

\setcounter{result}{0}

\newcommand{\sumn}[1]{%
\forloop{r}{1}{\value{r} < #1}%
{%
\setcounter{t}{\value{r}*\value{r}*\value{r}*\value{r}}
\setcounter{result}{\value{result}+\value{t}}
\theresult
}%
```

Рис. 4. Обчислення і друк суми введеної кількості натуральних чисел, піднесених до кубу

У тілі документа LaTeX викличемо команду `\sumn` і введемо кількість натуральних чисел більшу на одиницю (рис. 5). Якщо необхідно обчислити суму 120 натуральних чисел, піднесених до кубу, то введемо число 121 в якості аргумента команди `\sumn`.

```
\begin{document}

The sum of \((120\backslash)\) natural numbers raised to the power of
3 is \sumn{121}.

\end{document}
```

Рис. 5. Звертання до створеної команди у тілі документа

Наведений програмний код на рис. 4 і рис. 5 генерує результат, представлений на рис. 6.

```
The sum of 120 natural numbers raised to the power of 3 is
52707600.
```

Рис. 6. Результат, згенерований програмним кодом

Таким чином, границя заданої послідовності з загальним членом

$$a_n = 1^a + 2^a + 3^a + \dots + u^a + \left(\frac{1}{u}\right)^{2n}$$

при  $a=3$ ,  $u=120$ ,  $n \rightarrow \infty$ , дорівнює 52707600.

Створимо кореляційну таблицю, наприклад, з шести рядків і шести стовпців.

Для цього у LaTeX документі `\article` підключимо пакети `\pgf` і `\bnumexpr`. Створимо нову команду `\ifthenrow`, що містить сім аргументів, які можна ввести. Ця команда друкує рядки таблиці за умови, що введене число (перший аргумент цієї команди) дорівнює одиниці. Якщо введене число (перший аргумент цієї команди) не дорівнює одиниці, то рядки таблиці не друкуються. Решта аргументів команди `\ifthenrow` – це перший, другий, ..., шостий елементи відповідного рядка таблиці (рис. 7).

```
\documentclass{article}
\usepackage{pgf,bnumexpr}

\newcommand{\ifthenrow}[7]{\ifnum \bnumexpr#1\relax=1
#2&#3&#4&#5&#6&#7\\\fi
}
```

Рис. 7. Друк рядків таблиці за умови, що введене число (перший аргумент цієї команди) дорівнює одиниці

Якщо необхідно створити кореляційну таблицю з іншою кількістю рядків і стовпців, можна змінити число аргументів команди `\ifthenrow` на необхідну кількість.

У тілі документа LaTeX створимо таблицю і викличемо команду `\ifthenrow`, наприклад, у третьому рядку таблиці, ввівши одиницю в якості першого аргумента (рис. 8). У цьому випадку третій рядок таблиці надрукується.

```
\begin{document}

\begin{tabular}{|c|c|c|c|c|c|}

\hline
\(\xi \backslash \eta\) & 16 & 26 & 36 & 46 & 56 \\
\hline
20 & 4 & 0 & 0 & 0 & 0 \\
25 & 6 & 8 & 0 & 0 & 0 \\
\ifthenrow{1}{30}{0}{10}{32}{4}{0}
35 & 0 & 0 & 3 & 12 & 1 \\
40 & 0 & 0 & 9 & 6 & 5 \\
\hline

\end{tabular}

\end{document}
```

Рис. 8. Звертання до створеної команди у тілі документа

Наведений програмний код на рис. 7 і рис. 8 генерує результат, представлений на рис. 9.

$\xi \backslash \eta$	16	26	36	46	56
20	4	0	0	0	0
25	6	8	0	0	0
30	0	10	32	4	0
35	0	0	3	12	1
40	0	0	9	6	5

Рис. 9. Результат, згенерований програмним кодом

Якщо викликати команду `\ifthenrow` у третьому рядку таблиці і ввести, наприклад, нуль в якості першого аргумента (рис. 10), то третій рядок таблиці не надрукується.

```

\begin{document}

\begin{tabular}{|c|c|c|c|c|c|}

\hline
\(\xi \backslash \eta\) & 16 & 26 & 36 & 46 & 56 \\
\hline
20 & 4 & 0 & 0 & 0 & 0 \\
25 & 6 & 8 & 0 & 0 & 0 \\
\ifthenrow{0}{30}{0}{10}{32}{4}{0}
35 & 0 & 0 & 3 & 12 & 1 \\
40 & 0 & 0 & 9 & 6 & 5 \\
\hline

\end{tabular}

\end{document}

```

*Рис. 10. Звертання до створеної команди у тілі документа*

Наведений програмний код на рис. 7 і рис. 10 генерує результат, представлений на рис. 11.

$\xi \backslash \eta$	16	26	36	46	56
20	4	0	0	0	0
25	6	8	0	0	0
35	0	0	3	12	1
40	0	0	9	6	5

*Рис. 11. Результат, згенерований програмним кодом*

Обчислимо і надрукуємо суму елементів рядків і стовпців кореляційної таблиці, представленої на рис. 9.

Для цього у LaTeX документі `\article` підключимо пакети `\pgf` і `\bnumexpr`. Створимо нову команду `\sm`, що містить п'ять аргументів, які можна ввести. Ця команда обчислює і друкує суму введених аргументів. Наведемо програмний код на рис. 12.

```

\documentclass{article}
\usepackage{pgf, bnumexpr}

\newcommand{\sm}[5]{%
\the\numexpr#1+#2+#3+#4+#5\relax
}

```

*Рис. 12. Обчислення і друк суми введених аргументів*

Якщо необхідно створити кореляційну таблицю з іншою кількістю рядків і стовпців, можна змінити число аргументів команди `\sm` на необхідну кількість.

У тілі документа LaTeX створимо таблицю і виклинемо команду `\sm`, в якості аргументів якої введемо задані числа – елементи таблиці (рис. 13). Ця команда обчислить суму елементів рядків і стовпців.

Наведений програмний код на рис. 12 і рис. 13 генерує результат, представлений на рис. 14.

```

\begin{document}

\begin{tabular}{|c|c|c|c|c|c|c|}
\hline
\(\xi \backslash \eta\) & 16 & 26 & 36 & 46 & 56 &
sum\\
\hline
20 & 4 & 0 & 0 & 0 & 0 & \sm{4}{0}{0}{0}\\
25 & 6 & 8 & 0 & 0 & 0 & \sm{6}{8}{0}{0}\\
30 & 0 & 10 & 32 & 4 & 0 & \sm{0}{10}{32}{4}\\
35 & 0 & 0 & 3 & 12 & 1 & \sm{0}{0}{3}{12}{1}\\
40 & 0 & 0 & 9 & 6 & 5 & \sm{0}{0}{9}{6}{5}\\
\hline
sum & \sm{4}{6}{0}{0}{0} & \sm{0}{8}{10}{0}{0} & \sm{0}{0}{32}{3}{9} & \sm{0}{0}{4}{12}{6} & \sm{0}{0}{0}{1}{5} &
& \sm{4}{14}{46}{16}{20}\\
\hline
\end{tabular}

\end{document}

```

Рис. 13. Звертання до створеної команди у тілі документа

$\xi \backslash \eta$	16	26	36	46	56	sum
20	4	0	0	0	0	4
25	6	8	0	0	0	14
30	0	10	32	4	0	46
35	0	0	3	12	1	16
40	0	0	9	6	5	20
sum	10	18	44	22	6	100

Рис. 14. Результат, згенерований програмним кодом

**Висновки.** У статті виконано автоматизацію створення даних у математичних дослідженнях за допомогою створених нових макросів у системі LaTeX на мові програмування LaTeX. Створено нові макроси, які

- генерують матриці випадкових чисел, розмірності яких можна ввести на етапі виконання макросів;
- обчислюють і друкують суми введеної кількості доданків, піднесених до заданого степеня;
- друкують рядки введених таблиць при заданій умові;
- обчислюють і друкують суму елементів рядків і стовпців введених таблиць.

#### Список використаних джерел:

1. Gubal', G. N., & Stashenko, M. A. (2005). Improvement of an estimate of the global existence theorem for solutions of the Bogoliubov equations. *Theoretical and mathematical physics*, 145, 1736-1740.
2. Stashenko, M. A., & Gubal', G. N. (2006). Existence theorems for the initial value problem for the Bogolyubov chain of equations in the space of sequences of bounded functions. *Siberian Mathematical Journal*, 47, 152-168.
3. Hubal, H. M. (2012). The generalized kinetic equation for symmetric particle systems. *Mathematica Scandinavica*, 140-160.
4. Gubal', G. N. (2014). On the existence of weak local in time solutions in the form of a cumulant expansion for a chain of Bogolyubov's equations of a one-dimensional symmetric particle system. *Journal of Mathematical Sciences*, 199, 654-666.
5. Gryshchenko, T.V., Deineko, Zh.V. & Nikitenko, O.M. (2019). Using the LaTeX system during the preparation of scientific publications. *IV International Scientific and Technical Conference "Print,*

*Multimedia & Web*”, 96–98.

6. Губаль, Г. М. (2013). LATEX як видавнича система для створення математичних текстів і для програмування. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*, (12), 23-26.

7. Губаль, Г. М. (2013). Анімація в математичних текстах на мові LATEX. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*, 11, 11-15.

8. Lamport, L. (1994). *LaTeX: a document preparation system*. 2<sup>nd</sup> ed. Boston: Addison-Wesley. (272 p.).

9. Губаль, Г. М. (2013). Стратегії для створення математичної статті у видавничій системі LATEX. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*, (13), 10-13.

10. Губаль, Г. М. (2014). Особливості створення інтерактивних математичних тестів у видавничій системі LATEX. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*, (15), 9-13.

11. Lode, C. (2019). *Better books with LaTeX the agile way*. Clements Lode Verlag E.K. (254 p.).

12. Hubal, H.M. (2023) Improvement of references and footnotes in mathematical and other texts by creating macros in the LaTeX programming language. *International Journal on Information Technologies & Security*, (15(3)), 15–22.

13. Hubal, H. (2023). LATEX system in distance learning of the probability theory. *Grail of Science*, (33), 244-248.

14. Hubal, H. (2024). Analysis of Approaches to the Study of Limits of Sequences and Functions and the Use of Information Technologies. *Grail of Science*, (35), 226-231.

15. Hubal, H. (2024). Optimization of the Use of Some Modules in the Moodle System. *Grail of Science*, (38), 226-231.

16. Kottwitz, S. (2021). *LaTeX beginner's guide*. Packt Publishing, USA, (354 p.).

17. Kottwitz, S. (2023). *LaTeX graphics with TikZ*. Packt Publishing, USA, (304 p.).

18. Romansky, R. (2023). Empirical evaluation of the transfer of information resources in active learning. *International Journal on Information Technologies & Security*, 15(1), 39–48.

19. Hubal, H.M. (2019) Mathematical description of the non-equilibrium state of symmetric particle systems. *International Journal of Applied Mathematics*, (32(5)), 767–774.

20. Hubal, H.M. (2021) Mathematical modeling of biochemical processes rates in biological systems. *Computer-Integrated Technologies: Education, Science, Production*, (42), 43–49.

21. Van Dongen, M.R.C. (2012). *LaTeX and friends*. Springer-Verlag Berlin and Heidelberg GmbH & Co. K. (300 p.).

22. Verdaguer-Codina, J. (2022). Linux, LaTeX, and Python in secondary and baccalaureate. *Proceedings of IEEE Global Engineering Education Conference (EDUCON)*, Tunisia, 763–768.

23. Hubal, H. (2021). Mathematical modeling of biochemical processes rates in biological systems. *Computer-Integrated Technologies: Education, Science, Production*, (42), 43-49.

24. Губаль, Г. М. (2022). Математичне дослідження стійкості особливих точок систем диференціальних рівнянь, які описують швидкості біохімічних процесів. *Наукові нотатки*, (73), 29-39.

**Рецензент:** Федосов С.А., доктор фізико-математичних наук, професор кафедри фізики та вищої математики Луцького національного технічного університету.