**Название статьи здесь *заглавными буквами (Alt+A)***

**Подзаголовок *заглавными буквами* (Необязательно)(Alt+S)**

Имя Фамилия1,\* Имя Фамилия2(Alt+A)

*1 Введите данные об авторе 1 (Alt+L)*

*2 Введите данные об авторе 2 (Alt+L)*

*\*Автор-корреспондент. Электронная почта:* [*author@example.com*](mailto:author@example.com) (Alt+C)

**АННОТАЦИЯ**

В шаблоне статьи Atlantis Press Proceedings есть множество определённых абзацных стилей, которые вы можете использовать и применять при написании статьи. Чтобы отформатировать свою аннотацию, используйте стиль Microsoft Word: [Аннотация]. Каждая статья должна включать аннотацию. Начните аннотацию с заголовка “Аннотация”, выделенного жирным шрифтом, далее следует абзац с обычным шрифтом 10 кегль. Не цитируйте ссылки в аннотации. Пожалуйста, не размещайте и не цитируйте таблицы и рисунки в аннотации.

***Ключевые слова:*** *Ключевые слова вы выбираете самостоятельно, при оформлении работы, разделяете их запятыми (“,”). Ключевое слово 1, Ключевое слово 2, Ключевое слово 3, Ключевое слово 4.*

1. ЗАГОЛОВОК ПЕРВОГО УРОВНЯ (ГЛАВА 1)

Основной текст начинается со стандартного заголовка первого уровня, например, ВВЕДЕНИЕ или любого другого заголовка, соответствующего содержанию и контексту. Заголовки первого уровня оформляются заглавными буквами. Скопируйте содержимое и замените его другими заголовками первого уровня в оставшемся тексте. Ссылки должны быть заключены в квадратные скобки [1]. За заголовками всегда должен следовать текст.

Этот шаблон, изменённый в MS Word 2007 и сохраненный как "Документ Word 97-2003" для ПК, предоставляет авторам широкие возможности форматирования, необходимые для подготовки электронных версий статей. Все стандартные детали оформления статьи были определены по трем причинам: (1) простота использования при форматировании отдельных докладов, (2) автоматическое соответствие электронным требованиям, что облегчает одновременное или последующее производство электронной продукции, и (3) соответствие единому стилю всех материалов сборника конференции.

* 1. ***Заголовок второго уровня (Глава 2)***

Сначала убедитесь, что у вас правильный шаблон для вашего формата бумаги. Этот шаблон был специально разработан для вывода на бумаге формата А4.

Меню “Стили” в этом шаблоне следует использовать для форматирования текста, если это необходимо. Выделите текст, который вы хотите оформить определенным стилем, а затем выберите соответствующее название стиля в меню "Стиль". Стиль настроит ваши шрифты и межстрочный интервал. Для выделения фрагментов текста используйте курсив, а не подчёркивание. Чтобы вставить изображение в Word, установите курсор в точку вставки и используйте команду Вставка | Рисунок | Из файла или скопируйте изображение в буфер обмена Windows.

*1.1.1. Заголовок третьего уровня (Заголовок 3)*

Заголовки могут быть пронумерованы или нет (“Введение 1” и “1.2 Пронумерованный заголовок уровня 2”), без знаков препинания в конце. Как показано в этом документе, начальный абзац после заголовка не имеет отступа.

*1.1.1.2. Заголовок четвертого уровня*

Это заголовок четвертого уровня. Вы можете использовать его, где это уместно.

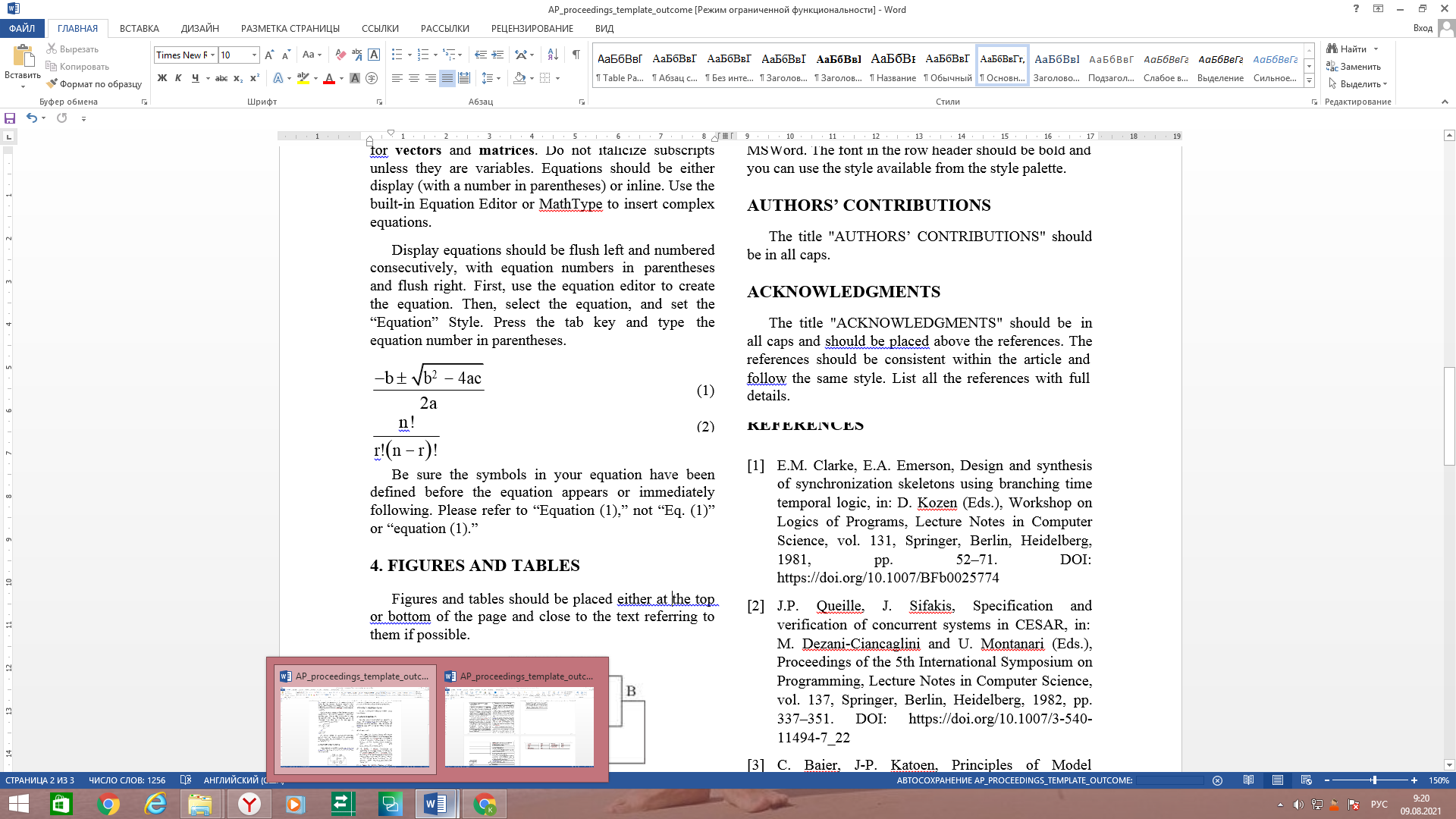
1. ПАЛИТРА СТИЛЕЙ

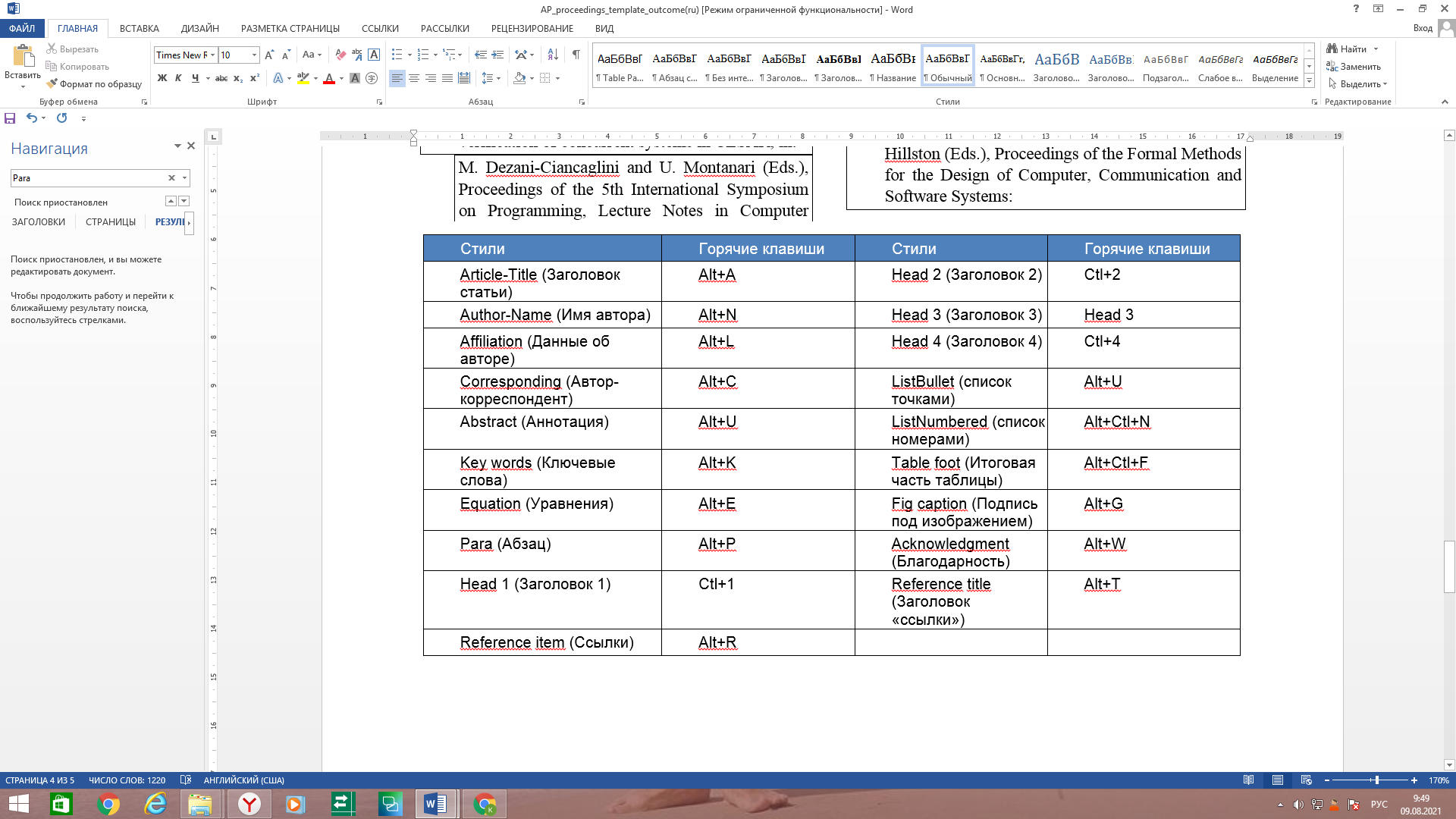
Стили можно применять с помощью палитры стилей, доступной в шаблоне. Чтобы активировать её, нажмите Ctrl+Shift+s. Применяйте стиль по мере необходимости, основываясь на содержании и контексте. (Пожалуйста, не выделяйте свой текст желтым цветом.)

1. ФОРМУЛЫ И УРАВНЕНИЯ

Простые *переменные* и *физические константы* должны быть выделены курсивом. Жирный (не курсив) шрифт должен применяться для **векторов** и **матриц**. Не выделяйте подстрочные индексы курсивом, если они не являются переменными. Уравнения должны быть либо отображаемыми (с числом в круглых скобках), либо встроенными. Используйте встроенный редактор уравнений или MathType для вставки сложных уравнений.

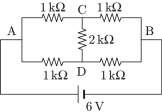
Отображаемые уравнения должны быть расположены слева и последовательно пронумерованы. Порядковые номера уравнений в круглых скобках расположены на одном уровне с уравнениями, но справа. Сначала воспользуйтесь редактором уравнений, чтобы создать уравнение. Затем выберите уравнение и установите стиль “Уравнение”. Нажмите клавишу tab и введите номер уравнения в круглых скобках.





Убедитесь, что символы в вашем уравнении были определены до формирования уравнения или сразу после. Пожалуйста, пишите “Уравнение (1)", а не “Уравнение. (1)” или “уравнение (1)”.

1. РИСУНКИ И ТАБЛИЦЫ

Рисунки и таблицы должны располагаться либо вверху, либо внизу страницы и, по возможности, рядом с текстом, на который они ссылаются.

**Рисунок 1** Содержание подписи. Название “Рисунок” и подпись должны быть выделены **жирным** шрифтом.

Для маленьких таблиц, пожалуйста, размещайте их в пределах одной колонки, а большие таблицы - в текстовой рамке, охватывающей обе колонки. Используйте таблицу, доступную в MSWord.

**Таблица 1.** Горячие клавиши для шаблона

Шрифт в заголовке строки должен быть жирным, и вы можете использовать стиль, доступный в палитре стилей.

АВТОРСКИЙ ВКЛАД

Заголовок "АВТОРСКИЙ ВКЛАД" должен быть написан заглавными буквами.

БЛАГОДАРНОСТЬ АВТОРА

Заголовок "БЛАГОДАРНОСТЬ АВТОРА" должен быть написан заглавными буквами и помещен над ссылками. Ссылки должны быть последовательными в рамках стать и соответствовать единому стилю. Перечислите все ссылки с полной информацией.

REFERENCES

1. E.M. Clarke, E.A. Emerson, Design and synthesis of synchronization skeletons using branching time temporal logic, in: D. Kozen (Eds.), Workshop on Logics of Programs, Lecture Notes in Computer Science, vol. 131, Springer, Berlin, Heidelberg, 1981, pp. 52–71. DOI: https://doi.org/10.1007/BFb0025774
2. J.P. Queille, J. Sifakis, Specification and verification of concurrent systems in CESAR, in:  
   M. Dezani-Ciancaglini and U. Montanari (Eds.), Proceedings of the 5th International Symposium on Programming, Lecture Notes in Computer Science, vol. 137, Springer, Berlin, Heidelberg, 1982, pp. 337–351. DOI: https://doi.org/10.1007/3-540- 11494-7\_22
3. C. Baier, J-P. Katoen, Principles of Model Checking, MIT Press, 2008.
4. M. Kwiatkowska, G. Norman, D. Parker, Stochastic model checking, in: M. Bernardo, J. Hillston (Eds.), Proceedings of the Formal Methods for the Design of Computer, Communication and Software Systems:   
   Performance Evaluation (SFM), Springer, Berlin, Heidelberg, 2007, pp. 220–270. DOI: https://doi.org/10.1007/978-3-540-72522-0\_6
5. V. Forejt, M. Kwiatkowska, G. Norman, D. Parker, Automated verification techniques for probabilistic systems, in: M. Bernardo, V. Issarny (Eds.), Proceedings of the Formal Methods for Eternal Networked Software Systems (SFM), Springer, Berlin, Heidelberg, 2011, pp. 53–113. DOI: https://doi.org/10.1007/978-3-642-21455-4\_3
6. G.D. Penna, B. Intrigila, I. Melatti, E. Tronci, M.V. Zilli, Bounded probabilistic model checking with the muralpha verifier, in: A.J. Hu, A.K. Martin (Eds.), Proceedings of the Formal Methods in Computer-Aided Design, Springer, Berlin, Heidelberg, 2004, pp. 214–229. DOI: <https://doi.org/10.1007/978-3-540-30494-4_16>
7. E. Clarke, O. Grumberg, S. Jha, et al., Counterexample-guided abstraction refinement, in:

E.A. Emerson, A.P. Sistla (Eds.), Computer Aided Verification, Springer, Berlin, Heidelberg, 2000,

pp. 154–169. DOI: <https://doi.org/10.1007/10722167_15>

1. H. Barringer, R. Kuiper, A. Pnueli, Now you may compose temporal logic specifications, in: Proceedings of the Sixteenth Annual ACM Symposium on the Theory of Computing (STOC), ACM, 1984, pp. 51–63. DOI: <https://doi.org/10.1145/800057.808665>
2. A. Pnueli, In transition from global to modular temporal reasoning about programs, in: K.R. Apt (Ed.), Logics and Models of Concurrent Systems, Springer, Berlin, Heidelberg, 1984, pp. 123–144.

DOI: <https://doi.org/10.1007/978-3-642-82453-1_5>

1. B. Meyer, Applying "Design by Contract", Computer 25(10) (1992) 40–51. DOI: <https://doi.org/10.1109/2.161279>
2. S. Bensalem, M. Bogza, A. Legay, T.H. Nguyen, J. Sifakis, R. Yan, Incremental component-based construction and verification using invariants, in: Proceedings of the Conference on Formal Methods in Computer Aided Design (FMCAD), IEEE Press, Piscataway, NJ, 2010, pp. 257–256.
3. H. Barringer, C.S. Pasareanu, D. Giannakopolou, Proof rules for automated compositional verification through learning, in Proc. of the 2nd International Workshop on Specification and Verification of Component Based Systems, 2003.
4. M.G. Bobaru, C.S. Pasareanu, D. Giannakopoulou, Automated assume-guarantee reasoning by abstraction refinement, in: A. Gupta, S. Malik (Eds.), Proceedings of the Computer Aided Verification, Springer, Berlin, Heidelberg, 2008, pp. 135–148. DOI: https://doi.org/10.1007/978-3- 540-70545-1\_