LIPIcs: Instructions for Authors and the `lipics-v2021` Class

Dagstuhl Publishing ✉ ⌂

Schloss Dagstuhl – Leibniz-Zentrum für Informatik
Oktavie-Allee, 66687 Wadern, Germany

─── **Abstract** ───

This document provides general instructions for authors on how to prepare articles to be submitted to the *Leibniz International Proceedings in Informatics* series and a description of the typesetting process. All LIPIcs articles are written in LaTeX using the LaTeX style class of the series. Starting in 2021, the newest version of this style (`lipics-v2021`) is used for publications. This document presents the functionalities of this style class, along with the instructions for the authors.

2012 ACM Subject Classification General and reference → Computing standards, RFCs and guidelines

Keywords and phrases LaTeX, type setting, style, changelog, `lipics-v2021`

Supplementary Material The style files including a sample document can be found here:
Software (Style): https://submission.dagstuhl.de/styles/download/lipics/authors
Software (Source Code): https://github.com/dagstuhl-publishing/styles/
    archived at swh:1:dir:1b66fc29b44e371e68a77879c0fbebefe3cd34f8

1 Typesetting instructions – Summary

LIPIcs – Leibniz International Proceedings in Informatics is a series of open access high-quality conference proceedings across all fields in informatics established in cooperation with Schloss Dagstuhl. In order to do justice to the high scientific quality of the conferences that publish their proceedings in the LIPIcs series, which is ensured by the thorough review process of the respective events, we believe that LIPIcs proceedings must have an attractive and consistent layout matching the standard of the series. Moreover, the quality of the metadata, the typesetting and the layout must also meet the requirements of other external parties such as indexing services, our DOI service provider, and funding agencies, among others. The guidelines contained in this document[1] serve as the baseline for authors, editors, and the publisher to create documents that meet as many of these requirements as possible.

Please comply with the following instructions when preparing your article.

Minimum Requirements

- Use pdflatex and an up-to-date LaTeX system[2].
- Use further LaTeX packages and custom made macros carefully and only if required.
- Use the provided sectioning macros: `\section`, `\subsection`, `\subsubsection`, `\paragraph`, `\paragraph*`, and `\subparagraph*` (for more details, see Section 2.4).
- Provide suitable graphics of at least 300dpi (preferably in PDF format).
- Use BibTeX and keep the standard style (`plainurl`) for the bibliography.
- Please try to keep the warnings log as small as possible. Avoid overfull `\hboxes` and any kind of warnings/errors with the referenced BibTeX entries.
- Use a spellchecker to correct typos.

───────────

[1] Since the mentioned requirements are constantly changing, these author guidelines but also the LIPIcs style are continuously adapted.

[2] Dagstuhl Publishing is currently using TeX Live 2021, see https://www.tug.org/texlive/.

Mandatory Metadata Macros

Please set the values of the metadata macros carefully since the information parsed from these macros will be passed to publication servers, catalogues and search engines. Avoid placing macros inside the metadata macros. For a more detailed explanation and a full list of all metadata macros, see Section 2.3. The following metadata macros/environments are mandatory:

- `\title` and, in case of long titles, `\titlerunning`.
- `\author`, one for each author, even if two or more authors have the same affiliation.
- `\authorrunning` and `\Copyright` (concatenated author names)
  The `\author` macros and the `\Copyright` macro should contain full author names (especially with regard to the first name), while `\authorrunning` should contain abbreviated first names.
- `\ccsdesc` (ACM subject classification[3]).
- `\keywords` (a comma-separated list of keywords).
- `\relatedversiondetails` (if there is a related version like a "Full Version", "Extended Version", or also a predecessor); please make sure to provide a persistent URL, e. g., at arXiv or HAL.
- `\begin{abstract}...\end{abstract}` .

Please do not . . .

Generally speaking, please do not override the `lipics-v2021`-style defaults. To be more specific, a short checklist also used by Dagstuhl Publishing during the final typesetting is given below. In case of **non-compliance** with these rules Dagstuhl Publishing will remove the corresponding parts of LaTeX code and **replace it with the `lipics-v2021` defaults**. In serious cases, we may reject the LaTeX-source and expect the corresponding author to revise the relevant parts.

- Do not use a different main font. (For example, the `times` package is forbidden.)
- Do not alter the spacing of the `lipics-v2021.cls` style file.
- Do not use `enumitem` and `paralist`. (The `enumerate` package is preloaded, so you can use `\begin{enumerate}[(a)]` or the like.)
- Do not use "self-made" sectioning commands (e. g., `\noindent{\bf My Paragraph}`).
- Do not hide large text blocks using comments or `\iffalse ... \fi` constructions.
- Do not use conditional structures to include/exclude content. Instead, please provide only the content that should be published – in one file – and nothing else.
- Do not wrap figures and tables with text. In particular, the package `wrapfig` is not supported.
- Do not change the bibliography style. In particular, do not use author-year citations. (The `natbib` package is not supported.)

This is only a summary containing the most relevant details. Please read the complete document for all details and don't hesitate to contact Dagstuhl Publishing (`publishing @dagstuhl.de`) in case of questions or comments.

Upholding the above-mentioned minimum formatting requirements will make the job of authors, volume editors, and Dagstuhl Publishing staff easier and will lead to the timely publication of volumes. There is no reason to tweak anything to save space in the published version of the papers.

---

[3] `https://submission.dagstuhl.de/services/acm-subject-classification`

Contents

2   General Instructions for Authors

On the following pages, the most important instructions for authors are provided to prepare a document to be published in a *Leibniz International Proceedings in Informatics* series volume.

2.1   Preparation

To prepare an article for the LIPIcs series, the authors have to provide a LaTeX[4] document using the `lipics-v2021` style. An alternative workflow using another format like Microsoft Word is not supported. Along with LaTeX, BibTeX has to be used to prepare the bibliography (see Section 2.10).

We suggest to employ a recent, up-to-date TeX installation: the most important distributions are TeX Live [5], MiKTeX/proTeXt and MacTeX. Older versions should (in principle) work as well. However, authors should not use deprecated commands. For instance, `\emph{...}` should be used instead of `{\em ...}`, see also [1].

~~\em~~ ~~\bf~~ ~~\tt~~
~~\sf~~ ~~\sc~~ ~~\it~~
→
`\emph \textbf`
`\texttt`
`\textsf`
`\textsc`
`\textit`

Tip: In some cases, the authors might want to have a consistent, uniform formatting of specific words or letters (e.g. variable or parameter names) independent of the formatting of the surrounding text. In this case, it is recommended to define a macro for this variable fixing the formatting, e.g. `\newcommand{\X}{\textup{\textbf{X}}}`.

2.2   The `lipics-v2021` Class

The `lipics-v2021` class assists authors in preparing articles for *Leibniz International Proceedings in Informatics* with LaTeX. It adapts LaTeX's standard `article` class to meet some requirements for LIPIcs and provides a specific layout.

The package[6] consists of the following files:

- `lipics-v2021-authors-guidelines.pdf` (this documentation),
- `lipics-v2021-sample-article.tex` (the LaTeX master file, to be used as a template),
- `cc-by.pdf` (logo for Creative Commons CC BY),
- `lipics-logo-bw.pdf` (logo for LIPIcs),
- `orcid.pdf` (logo for ORCID), and
- `lipics-v2021.cls` (the LaTeX class file, providing adaptions of `articles.cls` for LIPIcs and producing the layout).

To use `lipics-v2021`, put `lipics-v2021-sample-article.tex`, `lipics-v2021.cls`, `cc-by.pdf`, `lipics-logo-bw.pdf`, and `orcid.pdf` in your working directory, edit the file `lipics-v2021-sample-article.tex` in your preferred text editor and run LaTeX as usual. (See the following section for more detailed advise.)

`lipics-v2021` tries to benefit as far as possible from standard LaTeX packages[7]. Therefore, it should also be easy to compile an already written manuscript with the `lipics-v2021` layout. To learn more about the underlying packages, we refer to their documentations (try e.g., `texdoc [package name]` at your shell prompt or visit `tug.ctan.org`).

---

[4] This documentation is not intended to give an introduction to LaTeX. For questions concerning TeX systems/installations or the LaTeX mark-up language in general please visit `www.tug.org`, `www.dante.de`, `uk.tug.org` or any other TeX user group worldwide.
[5] Dagstuhl Publishing is currently using TeX Live 2020, see `https://www.tug.org/texlive/`.
[6] Available on `https://submission.dagstuhl.de/styles/download/lipics/authors`.
[7] A list of the used packages is provided in section A in the Appendix.

To ensure their long-term preservation, documents with the `lipics-v2021` style are generated in accordance with the PDF/A-3b standard[8]. A PDF/A file is completely self-contained. It includes everything needed for accurate representation of the contents. For this reason, the fonts in all used PDF files (e.g. figures) have to be embedded in them. Furthermore, PDF/A doesn't allow mixing color spaces. The default color space in `lipics-v2021` is the sRGB color space. So please don't use other color spaces. The PDF/A support is disabled by default to avoid problems with outdated LaTeX installation. To enable the support, add the option `pdfa` as argument of `\documentclass`. This option will be set during the finalization of the document.

`pdfa`

2.2.1 Paper Format

For LIPIcs, the A4 format is the mandatory paper format and will be used to prepare the final version. The option `a4paper` must be inserted in the optional argument of `\documentclass`. For intermediate versions, the US-letter format `letterpaper` can be used alternatively.

`\documentclass`
`a4paper`
`letterpaper`

Tip: You can add the option `draft` as argument of `\documentclass` to highlight overfull hboxes. This will also temporarily replace figures with placeholders, which might speed up the compilation process.

`draft`

Options: Table 1 shows the most common options for arguments of `\documentclass`. Some of the options are explained in later sections: `USenglish` (Section 2.2.2), `UKenglish` (Section 2.2.2), `numberwithinsect` (Section 2.7), `cleveref` (Section 2.4), `autoref` (Section 2.4), `anonymous` (Section 2.3), and `authorcolumns` (Section 2.3).

Table 1 Common Options for `\documentclass`.

Option	Description
`a4paper`	A4 paper format
`letterpaper`	US-letter format
`draft`	Highlight overfull hboxes
`USenglish`	Enables US-English specific typographical rules
`UKenglish`	Enables UK-English specific typographical rules
`numberwithinsect`	Enables section-numbered lemma etc.
`cleveref`	Enables support for the `cleveref` package
`autoref`	Enables support using `autoref` provided by the `hyperref` package
`anonymous`	Enables support for double-blind peer-review
`thm-restate`	Enables support for the `thm-restate` package for restating theorems
`authorcolumns`	Enables two-column displaying of author details (only applicable for more than 6 authors)
`pdfa`	Enables the generation of a PDF according to the PDF/A standard

---

[8] See https://www.pdfa.org/

2.2.2 Language

USenglish
UKenglish

The document language is chosen in the optional argument of the `\documentclass` command in the LaTeX master file. This is necessary to influence the "culturally-determined typographical (and other) rules" (like hyphenation) within the document. Possible values are USenglish, UKenglish and others[9].

Tip: Older version of the used LaTeX package `babel` might not know the default language (UKenglish). For many installations, setting `english` as the document language solves the problem: `\documentclass[english]{lipics-v2021}`. Please note that out-dated language options such as `english` will be replaced by the LIPIcs default UKenglish during the finalization process.

2.2.3 Input Encoding

`lipics-v2021` preselects UTF-8 as input encoding. Please do not change the input encoding because otherwise the volume compilation might become difficult.

2.2.4 Fonts

`lipics-v2021` uses the Latin Modern font family. This is a recent redesign of the good old Computer Modern fonts. Latin Modern provides many characters and all necessary mathematical fonts. If your TeX installation does not provide the Latin Modern family, Computer Modern is used as a fallback. Do not use a different main font. For example, the usage of the `times` package is forbidden.

MnSymbol

`lipics-v2021` preloads the package `amssymb` to make additional mathematical symbols available. Other symbol packages, e.g., `stmaryrd`, may be added. Moreover, the script mathematical alphabet is provided by loading the `eucal` package. Please do not use the `MnSymbol` package, as it is apparently not possible to create PDF/A-compliant PDFs with this package[10].

To ensure the PDF/A compliance, please make sure that all fonts are embedded in all used PDF files (e.g. figures).

2.2.5 Colors

In the LIPIcs style, very few colours are used for purely aesthetic purposes. To increase readability especially for color-blind persons, we recommend not to use colors for emphasizing text. Instead of this, other mechanisms like `\textbf`, `\emph` or dashed boxes (e.g. provided

\dashbox

by `dashbox`[11] package) can be used.

Table 2 lists all colors that are pre-defined and used in `lipics-v2021`. To reuse one of the prefined colors, just use the name given in the first column of the table (e.g. `\textcolor{lipicsGray}`).

---

[9] This depends on the LaTeX package `babel`. For further details, please visit the documentation of the package at `https://www.ctan.org/pkg/babel`.
[10] See `https://tex.stackexchange.com/questions/57292/pdf-a-width-information-for-rendered-glyphs-is-inconsistent`.
[11] `https://ctan.org/pkg/dashbox`

Table 2 Pre-defined colors in `lipics-v2021`.

Name	Color	rgb Code	Usage
`lipicsGray`		0.31, 0.31, 0.33	heading in top matter, enumerations
`lipicsBulletGray`		0.60, 0.60, 0.61	bullet points, footnoterule
`lipicsLineGray`		0.51, 0.50, 0.52	line in heading of abstract and bibliography
`lipicsLightGray`		0.85, 0.85, 0.86	background of listings
`lipicsYellow`		0.99, 0.78, 0.07	section numbers, figure and table captions

To ensure the PDF/A compliance, please only use colors from the sRGB color space. The usage of colors from the CMYK color space is not supported as PDF/A doesn't allow the mix of color spaces.

2.2.6 Spacing and Page Length

`lipics-v2021` provides a liberal spacing to improve the readability of the articles. Please do not modify the default spacing and also do not use any packages (e.g. `savetree`) to manipulate the LIPIcs default spacing. It will be reset to default during the final typesetting process by the LIPIcs office. To avoid any problems due to the LIPIcs liberal spacing, it is highly recommended for the editors of LIPIcs volumes to set a flexible page limit[12] (e. g., of 12–15 pages) and to explicitly exclude the bibliography, the first page(s) containing the top matter (authors, affiliation, keywords, abstract, . . .) and a brief appendix (of up to 5 pages) from this page limit.

~~savetree~~

During the preparation, the authors should take care of suitable linebreaks and pagebreaks. For instance, no overfull `\hboxes` should occur in the warnings log.

`\hbox`

2.2.7 Main Elements in the Document

The body of a LIPIcs document consists in general of the following elements:
- top matter containing the metadata describing the document (see Section 2.3);
- section headings (see Section 2.4);
- figures and tables (see Section 2.5);
- lists (see Section 2.6);
- theorem-like environments like lemmas, definitions, remarks, and proofs (see Section 2.7);
- mathematics, both in text and in display style (see Section 2.8);
- listings and algorithms (see Section 2.9);
- references (see Section 2.10);
- and text.

2.3 Top Matter: Authors, Affiliations, and other Metadata

The term "top matter" is used in this document for the metadata information found at the beginning of an article. It includes information that would appear in a bibliographic reference to the paper, plus information to clearly identify the authors, funding information,

---

[12] Please note that we expect a certain flexibility at any page limit (+/- 1 page) to avoid too much extra effort to harmonize LIPIcs-compliant formatting and page limit for both authors and final typesetting by the LIPIcs Office.

acknowledgements, and the abstract. The top matter also includes information such as subject classification and keywords that simplifies the indexation of the articles. Some elements are required, while others are optional (each element is described in detail below).

The header of a LIPIcs article is the only part where some specific commands/environments are required:

`\title`
`\titlerunning`

- The title is tagged as usual with the `\title{...}` command. If you need a short form for the running head, use the optional `\titlerunning{...}`. The title (as well as headings in the document) should be capitalized (see Section 2.4 for further details).

`\author`

- Authors and their affiliations are rendered blockwise for LIPIcs[13]. The information for each author has to be tagged with a redefined `\author` macro, which is structured as follows: `\author{name}{affil}{email}{orcid}{funding}`. While the parameters `name` and `affil` are mandatory, the parameters `email`, `orcid`, and `funding` are optional and can be empty.

- For each author, at least one affiliation has to be provided. It is up to the author to decide about the level of detail of the affiliation information, but at least the main affiliation name, the city[14], and country have to be provided. Further details like department or postal address are optional. If several affiliations should be provided for one author,

`\and`
please use the `\and` macro to separate the different affiliations. Authors without an affiliation should at least indicate their place of residence and their country of residence. An (optional) URL of an author's personal webpage should simply be added to the affiliation with `\and`, for instance

```
\author{John Q. Public}
{Dummy University Computing Laboratory, [optional: Address], Country \and
My second affiliation, Country \and \url{http://www.myhomepage.edu}}
{johnqpublic@dummyuni.org}
{https://orcid.org/0000-0002-1825-0097}
{[optional] author-specific funding acknowledgements}
```

To indicate the correct name splitting when the last name consists of multiple name parts, it is possible to use an additional pair of curly braces around the last name, e.g. `\author{Jane {Open Access}}{...}{}{}{}`.

- The usage of an ORCID is currently optional but highly recommended. For more details regarding ORCID, please visit `https://orcid.org/`. If you don't have an ORCID yet, the registration is very quick and simple (see `https://orcid.org/register`). Please enter the full https URL including hyphens between every fourth digit as recommended by ORCID[15]. In the PDF, the ORCID symbol that links to the ORCID profile is shown behind the respective author. With the next upgrade of *Dagstuhl Research Online Publication Server (DROPS)*, ORCIDs will also become part of the machine readable metadata set.

- Per default, author details (name, affiliation, . . .) are displayed in one column under each other. In case of many authors, this might result in a lot of whitespace (especially if the authors have only short affiliations). To avoid this, it is possible to activate a two column

`authorcolumns`
style for the authors by adding `authorcolumns` as argument of the `\documentclass`: `\documentclass[authorcolumns]{lipics-v2021}`. Please note that this option is only available in case of more than 6 authors.

---

[13] This has changed in 2017; the formerly used `authblk` mechanism to output authors and affiliations in footnote style is deprecated; the `\affil` command is obsolete for the lipics-v2021 class.

[14] The city is only mandatory if it is not part of the affiliation name.

[15] `https://support.orcid.org/knowledgebase/articles/116780`

- There is a mandatory `\authorrunning{...}` macro for the running header concatenating all author names (first names have to be abbreviated).

  `\author-running`

- `\Copyright{...}` has just one argument specifying the copyright holder that is usually the team of authors. Please concatenate the full names of all authors[16].

  `\Copyright`

- To hide all author related information, e.g., for double-blind peer review, please add `anonymous` as argument of the `\documentclass`:
  `\documentclass[anonymous]{lipics-v2021}`.

  `anonymous`

- `\ccsdesc{...}` is for classification information following the ACM 2012 Computing Classification System; it is required, too. Please use one `\ccsdesc` command per classification. If the command is used several times, all specified classifications are concatenated in the resulting PDF (using semicolons as auto-generated delimiters). The `\subjclass` command used in previous versions of the LIPIcs LATEX style is obsolete. Please choose 1–3 ACM 2012 classifications from `https://submission.dagstuhl.de/services/acm-subject-classification`.

  `\ccsdesc`

- `\keywords{...}` must be used to capture keywords. Please use commas as delimiters. The first word of the list and every proper noun should be capitalized. All other items should be lowercase.

  `\keywords`

- `\category{...}` may be used to provide category information. This macro should only be used for "special" papers (like "invited papers") and not for regular articles. Please consult the responsible editor of the volume.

  `\category`

- `\relatedversiondetails{...}` may be used to denote a related version like a "Full Version", "Extended Version", or also a predecessor usually published in a repository like arXiv or HAL. The macro has two mandatory parameters and two optional arguments: `\relatedversiondetails[linktext={...}, cite=...]{Classification}{URL}` to be used as follows:

  `\related-version-details`

  - `Classification`: Classification of the related version, e.g. "Extended Version", "Full Version", or any free text.
  - `URL`: Persistent URL of the related version[17].
  - `linktext`: This text is shown alternatively to the actual URL. Please use only to manually add linebreaks to the URL.
  - `cite`: BibTEX key of the related version.

  The macro `\relatedversion{...}` may be used to provide an additional textual explanation, which might especially be useful in case of several related documents. Figure 1 shows an example for such a use case. The corresponding LATEX code is depicted in Listing 1.

  `\related-version`

Related Version This paper is based on our following two technical reports:
Previous Version: `https://arxiv.org/abs/XYZ.ZYX1` [2]
Previous Version: `https://arxiv.org/abs/XYZ.ZYX2` [1]

■ **Figure 1** Example for `\relatedversiondetails` and `\relatedversion` usage.

---

[16] Please add "serial comma" also know as "Oxford comma" before "and".

[17] Please note: As hosting on a (personal or university) webpage or in cloud storage is not really sufficient for durable / persistent file storage, we highly recommend you to publish your document in a reliable repository like arXiv or HAL. This recommendation also applies to supplementary materials such as data, software, posters, ..., slides. Please store them in repositories such as zenodo, figshare, or Software Heritage.

■ **Listing 1** LATEX code of Figure 1.

```
\relatedversion{This paper is based on our following
        two technical reports:}
\relatedversiondetails[cite=DBLP:books/mk/GrayR93]{Previous Version}
        {https://arxiv.org/abs/XYZ.ZYX1}
\relatedversiondetails[cite=DBLP:journals/cacm/Dijkstra68a]
        {Previous Version}{https://arxiv.org/abs/XYZ.ZYX2}
```

`\supplement-`
`details`

▬ `\supplementdetails{...}` may be used to denote supplementary material like related research data, source code, . . . slides hosted on a repository[17] like zenodo[18], figshare[19], . . . , Software Heritage[20]. The macro has two mandatory parameters and four optional arguments: `\supplementdetails[linktext={...}, cite=..., subcategory={...}, swhid=...]{Classification}{URL}` to be used as follows:

  ▬ `Classification`: Classification of the supplementary material using a controlled vocabulary. Allowed values are `Audiovisual`, `Collection`, `DataPaper`, `Dataset`, `Event`, `Image`, `InteractiveResource`, `Model`, `PhysicalObject`, `Service`, `Software`, `Sound`, `Text`, `Workflow`, and `Other`[21].
  ▬ `URL`: Persistent URL of the supplementary material[17].
  ▬ `linktext`: This text is shown alternatively to the actual URL. Please use only to manually add linebreaks to the URL.
  ▬ `cite`: BIBTEX key of the supplementary material.
  ▬ `subcategory`: Free text subcategory to further specify the supplementary material .
  ▬ `swhid`: Software Heritage Identifier[22] of the archived supplementary material. This is especially recommended for software[23].

`\supplement`   `\supplement{...}` may be used to provide an additional textual explanation, which might especially be useful in case of several supplementary materials. Figure 2 shows an example for such a use case. The corresponding LATEX code is depicted in Listing 2.

Supplementary Material ECRTS 2019 Artifact Evaluation approved artifact and the source code for the Dagstuhl Publishing styles used to prepare this document can be found as follows:
Software (ECRTS Artifact Evaluation approved artifact): `https://doi.org/10.4230/DARTS.5.1.0`
Software (Source Code): `https://github.com/dagstuhl-publishing/styles/`
  archived at `swh:1:dir:1b66fc29b44e371e68a77879c0fbebefe3cd34f8`

■ **Figure 2** Example for `\supplementdetails` and `\supplement` usage.

■ **Listing 2** LATEX code of Figure 2.

```
\supplement{ECRTS 2019 Artifact Evaluation approved artifact and
        the source code for the Dagstuhl Publishing styles used
        to prepare this document can be found as follows:}
\supplementdetails[subcategory={ECRTS Artifact Evaluation approved
        artifact}]{Software}{https://doi.org/10.4230/DARTS.5.1.0}
\supplementdetails[subcategory={Source Code},
        swhid={swh:1:dir:1b66fc29b44e371e68a77879c0fbebefe3cd34f8}]
        {Software}{https://github.com/dagstuhl-publishing/styles/}
```

---

[18] `https://zenodo.org/`
[19] `https://figshare.com/`
[20] `https://www.softwareheritage.org/`
[21] For detailed description of these types, please see the appendix of the DataCite Metadata Schema `https://doi.org/10.14454/7xq3-zf69`
[22] See `https://docs.softwareheritage.org/devel/swh-model/persistent-identifiers.html`.
[23] The Dagstuhl Submission System will also provide extended support for this in the near future.

▬ \funding{...} may be used to capture a general funding statement. Note that we \funding
recommend to provide the kinds of funding acknowledgements that can be clearly assigned
to a specific author in the corresponding \author macro. In the PDF document, all the
funding information is displayed in a coherent block, where the author-specific details
will automatically be preceded by the name of the author.

To display a flag or logo near the funding information as requested by some funding
agencies (e.g. ERC grant), please use the \flag command within the funding statement,
e.g. \flag{ERCandEU.pdf}[24].

▬ \acknowledgements{...} may be used to capture an acknowledgement. \acknowledge-
ments

The commands mentioned so far should be used in the document preamble of the LaTeX
file. Providing a title, at least one author (incl. \authorrunning), copyright information,
subject classification, and keywords is required.

To typeset an abstract use \begin{abstract}...\end{abstract} . The environment abstract
must be placed after \begin{document} and \maketitle!

Note that subject classifications, keywords, . . . , and acknowledgements will be rendered
together with the abstract. So it is necessary to use the abstract environment in order to
get the output for keywords, ccsdesc etc.

All the copyright information (and a possible DOI specification) on the first page can
be hidden using the \hideLIPIcs command. This might be useful to prepare a document \hideLIPIcs
using the LIPIcs style, but which is not intended to be published within the series (e.g. for
an extended version to be published on arXiv).

Hint. To ensure pdflatex processing, please add \pdfoutput=1 in the first lines of your \pdfoutput=1
document after the documentclass. This might avoid problems e.g. when uploaded your
document on arXiv.

2.4 Structuring the Document

To organize the content and to structure the document, the default LaTeX commands for \section
headings have to be used. At least the first three levels of section headings have to be numbered \subsection
to support the internal referencing of the respective sections and also to simplify referencing \subsubsection
in other documents. Table 3 provides an overview of the provided heading commands. All \paragraph
headings, including the title, have to be left aligned and should be capitalized. \paragraph*
\subparagraph

Please do not revise the provided commands and e. g., modify text size or spacings.
Moreover, please do not add self-defined commands to structure your document. The LIPIcs
office will reset self-defined heading styles to the default, for instance
\medskip\noindent\textbf{Some subparagraph.} will be replaced by
\subparagraph*{Some subparagraph.}

Capitalization. The title of the document has to be capitalized: The first word (and usually
the last word) is capitalized, as well as all major words. Minor words are in lower case.
For a detailed description, please use the Chicago Style Guideline or use a web service,
such as http://individed.com/code/to-title-case/. This way of capitalization is also
recommended for headings.

---

[24] Logos and flags are not part of the style package and have to be added manually. ERC logos can be
found here: https://erc.europa.eu/managing-your-project/communicating-your-research

■ **Table 3** Headings.

Command	Level	Example	Font size and style
`\title`	0	**This is the Title**	17.28pt, bold
`\section{...}`	1	**1 Section**	12pt, bold
`\subsection{...}`	2	**1.1 Subsection**	12pt, bold
`\subsubsection{...}`	3	**1.1.1 Subsubsection**	12pt, bold
`\paragraph{...}`	4	**1.1.1.1 Paragraph**	10.5pt, bold
`\paragraph*{...}`	4	**Paragraph**	10.5pt, bold
`\subparagraph*{...}`	5	**Subparagraph.** Some text after	10pt, bold

Referencing. LaTeX provides an easy mechanism to reference sections and also almost everything else that is numbered (e. g., figures, tables, algorithms, . . .). To use this mechanism,
`\label`
`\ref` all referenced items should have a marker (a.k.a. `\label{key}`) and instead of direct textual references, the command `\ref{key}` should be used within the LaTeX code.
`\cref`
`\autoref` It is up to the authors to use the extended functionalities provided by either the `cleveref`[25] package or the `\autoref{}` command, which is part of the `hyperref`[26] package. To use these functionalities, please add `cleveref` and/or `autoref` as argument of the `\documentclass`: `\documentclass[cleveref, autoref]{lipics-v2021}`. This is explicitly necessary for using these mechanisms with theorem-like environments. Otherwise all theorems, lemmas, corollaries, . . . will be referenced as theorems (see also Section 2.7).

2.5   Figures and Tables

`figure`
`table` Figures, tables, and all other floating objects (like algorithms) can be placed at the top or bottom of a document page (using the placement option `t` or `b`, e. g., `\begin{figure}[t]`). Alternatively they can be placed on a separate page using the placement option `p`. The top position is the preferred position. The fixed position in the document (placement option `h`) should only be used rarely as small changes in the document might result in a completely different layout.

~~`wrapfig`~~ Wrapping of text around figures and tables is not supported. So don't use packages like `wrapfig`.

`\caption`
`\label` Use captions, labels, and references for all floating objects explicitly (e. g., "`as depicted in Figure \ref{fig:example}`") and not implicitly (e. g., "`as depicted in the figure`
`\ref` `below`"). Figure captions have to be placed below the figures. Table captions (and also captions of other text-like floating environments like listings and algorithms) have to be placed above the table. Captions are always left aligned.

`\subcaption` For small/related figures, place figures as sub-figures within one figure. At least the
`subfigure` main figure should have a textual caption. The preloaded `subcaption`[27] package provides
`subtable` the `\subcaption` command to add sub-captions. Alternatively, it provides `subfigure` and `subtable` environments. Please do not load the deprecated `subfig` package. The following fragment shows the code resulting in Figure 3:

---

[25] https://ctan.org/pkg/cleveref
[26] https://ctan.org/pkg/hyperref
[27] https://ctan.org/pkg/subcaption

```
lipics-logo-bw.pdf
```

```
lipics-logo-bw.pdf
```

(a) Subfig 1.   **(b)** Subfig 2.
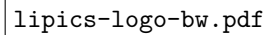
🟨 **Figure 3** Just an Example.

🟨 **Listing 3** LATEX code of Figure 3.

```
\begin{figure}
  \begin{subfigure}[t]{0.5\textwidth}
    \centering
    \includegraphics[draft, width=0.3\textwidth]{lipics-logo-bw}
    \caption{Subfig 1.}
  \end{subfigure}\hfill
  \begin{subfigure}[t]{0.5\textwidth}
    \centering
    \includegraphics[draft, width=0.3\textwidth]{lipics-logo-bw}
    \caption{Subfig 2.}
  \end{subfigure}
  \caption{Just an Example.}\label{fig:example}
\end{figure}
```

Figures and tables have to be centered. For figures and tables placed within a floating environment, use `\centering`. For figures and tables placed directly within the text, use the `center` environment.
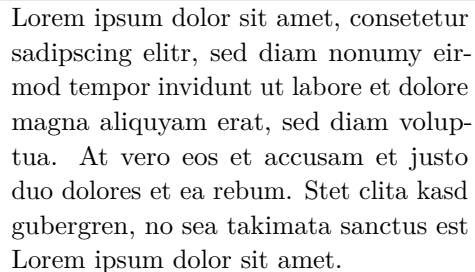
`\centering`
`center`

It is highly recommended to use vector graphics whenever possible. Alternatively, authors should provide figures with a high resolution (300 dpi or more).

To ensure PDF/A compliance of the resulting document, also embedded PDF files such as figures have to be completely self-contained. For this reason, the fonts in all used PDF files (e.g. figures) have to be embedded in them. Furthermore, PDF/A doesn't allow mixing color spaces. The default color space in lipics-v2021 is the sRGB color space. So please don't use other color spaces.

Hint. If your figure consists mainly of text, we highly recommend to frame the figure (e.g. using the command `\fbox`). This significantly helps to separate the content of the figure from the adjacent main text. Figure 4 shows an example for such a framed figure. The corresponding LATEX code is depicted in Listing 4.

`\fbox`

> Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

🟨 **Figure 4** Example for framed figure with mainly textual content.

🟨 **Listing 4** LATEX code of Figure 4.

```
\begin{figure}
\centering
\fbox{
\begin{minipage}{0.9\textwidth}
Lorem ipsum dolor sit amet ...
\end{minipage}
}
\caption{Example for framed ...}
\label{fig:framedExample}
\end{figure}
```

2.5.1 Graphics

`\include-`
`graphics`
`\graphicspath`

The standard interface for graphic inclusion is the `\includegraphics` command provided by the `graphicx` package. Note that the `\graphicspath` command allows authors to declare one or more folders in which the `graphicx` package looks for the image files; so providing the path with each `\includegraphics` command is not necessary.

2.5.2 Tables

Preloaded packages are: the `array`[28] package (for introducing new column types), the `multirow`[29] package (row spanning cells) and the `tabularx`[30] package (automatic column width calculation).

The `threeparttable` package is preloaded to ease the use of table footnotes. Please read the short documentation in `threeparttables.sty`[31] to see how the related commands are applied.

2.5.3 Rotating Floats

`sidewaysfigure`
`sidewaystable`

The preloaded `rotating` package provides the two environments `sidewaysfigure` and `sidewaystable`. They allow for the rotation of floating objects.

2.6 Lists, Enumerations, and Descriptions

`itemize`
`enumerate`
`description`

Other elements that provide structured information within LATEX documents are bulleted lists (also known as `itemize` environments), enumerations (a.k.a. `enumerate` environments), and descriptions. List labels are set flush left.

The `enumerate`[32] package is preloaded, so you can use `\begin{enumerate}[(a)]` or the like. `lipics-v2021` provides several predefined environments, namely `itemize`, `enumerate`, `alphaenumerate`, `romanenumerate`, `bracketenumerate`, and `description`. Examples of the different styles are depicted in Section B in the appendix.

~~`enumitem`~~
~~`paralist`~~

Please do not use other packages like `enumitem` or `paralist` to revise itemize or enumerate environments, as they change the predefined layout of all related environments and therefore will be removed by Dagstuhl Publishing in the final typesetting process.

2.7 Theorem-like Environments

`theorem`
`lemma`
`conjecture`
`corollary`
`definition`
`proposition`
`exercise`
`example`
`note`
`note*`
`observation`
`remark*`
`remark`
`claim`
`claim*`

To display theorem-like fragments appropriately, the `amsthm`[33] package is preloaded, and the following environments are already introduced: `theorem`, `lemma`, `conjecture`, `corollary`, `definition`, `proposition`, `exercise`, `example`, `note`, `note*`, `observation`, `remark*`, `remark`, `claim`, and `claim*`. For the predefined environments, a consecutive numbering is applied, e.g., all predefined theorem-like environments use the same counter. Examples of the different styles are depicted in Section C in the appendix.

Setting up additional environments works with the `\newtheorem` mechanism from the `amsthm` package. For example, add to your document preamble

`\newtheorem{conjecture}[theorem]{Conjecture}`

---

[28] `https://www.ctan.org/pkg/array`
[29] `https://www.ctan.org/pkg/multirow`
[30] `https://www.ctan.org/pkg/tabularx`
[31] `https://www.ctan.org/pkg/threeparttable`
[32] `https://www.ctan.org/pkg/enumerate`
[33] `https://www.ctan.org/pkg/amsthm`

See also the `amsthm` package documentation.

Available `\theoremstyle`s are: `plain`, `definition`, `remark`, and `claimstyle` (the first three are from the `amsthm` package, but slightly modified for LIPIcs).

Note that for LIPIcs, it is recommended that all numbered theorem-like environments use one and the same counter, i. e., the counter of the default environment `theorem`.

By default, theorem-like environments are numbered consecutively throughout the document. To number the environments subordinately within sections use the class option `numberwithinsect`: `\documentclass[numberwithinsect]{lipics-v2021}`.

number-
withinsect

Additional to the above mentioned numbered environments, `lipics-v2021` also provides two unnumbered proof environments: `proof` and `claimproof`. Both differ only in the delimiting QED symbol; ◄ is automatically appended at the end of `proof` environments; ◁ is appended at the end of `claimproof` environments. To add the QED symbol manually at a certain position (e.g. to avoid the QED symbol in a new, empty line after an enumeration) use `\qedhere` in a `proof` environment or `\claimqedhere` in a `claimproof` environment.

proof
claimproof

\qedhere
\claimqedhere

Hint 1. To improve the readability of long proofs, it may be necessary to structure these proofs additionally. For this purpose, it is strongly recommended to use `description` environments. Alternatively, the style provides a `\proofsubparagraph` command to add a subparagraph-like heading within the proof.

\proofsubparagraph

To repeat a theorem-like environment, the style supports the package `thm-restate`, which is provided as part of the `thmtools`[34] bundle. To activate this support, please add `thm-restate` as argument of the `\documentclass`:
`\documentclass[thm-restate]{lipics-v2021}`.

thm-restate

For further information, please see the `thmtools` documentation.

Hint 2. In difference to the style provided by `amsthm` package, the body of the `definition` environment is formatted as in `theorem` environments, e. g. italic text and not upright text as in the original environment. This should help to separate text of the definition from surrounding regular text. If an upright text as in the original environment is preferred, please add an alternative `definition` environment to the document preamble, e. g. as follows:

```
\theoremstyle{definition}
\newtheorem{definition2}[theorem]{Definition}
```

Hint 3. Depending on the style of the environment, it might be difficult to clearly identify the end of the respective environment, see the following example:

This is just some regular text right before an example environment.

▶ **Example 1.** This is the first paragraph of the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.
This is another paragraph of the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

This is regular text after the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

---

[34] `https://www.ctan.org/pkg/thmtools`

`\lipicsEnd`

In this case, some authors prefer to clearly mark the end of the environment. This can be done using the command `\lipicsEnd`. This will add a symbol ($\lrcorner$) at the end of the line (similar to a qed symbol), see Example 2 below:

▶ **Example 2.** This is the first paragraph of the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

This is another paragraph of the example. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.   ⌟

Hint 4. As mentioned previously, to use `cleveref` and/or `autoref` also with theorem-like environments, please add `cleveref` and/or `autoref` as argument of the `\documentclass`: `\documentclass[cleveref, autoref]{lipics-v2021}`.

2.8 Equations / Mathematical formulas

`\(...\)`
`$...$`
`equation`
`equation*`
`\[...\]`

Equations or mathematical formulas can either be in line with the text (using `\(...\)` or `$...$`) or in an equation-type environment (e. g., `\[ ... \]`). The latter use is also known as display math mode. In this display math mode, the separately displayed formulas should be numbered (consecutive numbering is recommended). Important equations (and all large equations) should be placed in a separate block (i.e. in display math mode) and not inline. Furthermore, important equations should be labeled to reference them.

The `amsmath` package is preloaded, and you are encouraged to use the mark-up it provides instead of old-style standards like the `eqnarray` environment or the `\over` command.

~~`$$...$$`~~

Display math formulas. Do not use the TEX primitive `$$...$$` to typeset display math formulas as this might cause problems[35], for instance the alignment option is ignored. Instead use the LATEX variant of this environment, namely `\[...\]`. During the final typesetting process, the LIPIcs office will replace `$$...$$` with `\[...\]`.

`\[...\]`

To avoid inconsistent spacing, do not use the `displaymath`, `eqnarray`, or `eqnarray*` environments because those are not supported by the preloaded `amsmath` package. `\[...\]` may be used instead of `displaymath`, while `eqnarray`, and `eqnarray*` may be replaced with `align`, or `align*`.

`align, align*`

Display math formulas are aligned left. So do not use environments that ignore/change this alignment of these formulas. Deviating from this rule, the centered setting of inference rules (e.g. using the `mathpartir`[36] package) is permitted, however, for reasons of consistency, a left-justified orientation is also recommended here.

2.9 Listings and Algorithms

An important part in many computer science publications is "pseudo code" or code fragments. In the LATEX ecosystem, several packages exist to format code/algorithms. It is highly recommended to use one of the following packages and not to use self-defined environments: `listings`[37], `algorithm`[38], or `algorithm2e`[39].

---

[35] For more details, visit `https://ctan.org/pkg/l2tabu-english`, [1].

[36] `https://www.ctan.org/pkg/mathpartir`

[37] `https://www.ctan.org/pkg/listings`

[38] `https://www.ctan.org/pkg/algorithms`

[39] `https://www.ctan.org/pkg/algorithm2e`

The `listings` package is preloaded. It provides the `lstlisting` environment to typeset displayed code. Here, the package is configured to produce a grey background for listings.

The following example shows how to use captions and labels with the `lstlisting` environment:

```
\begin{lstlisting}[caption={Useless code.},label=list:8-6,float=h,
                                    abovecaptionskip=-\medskipamount]
for i:=maxint to 0 do
begin
    j:=square(root(i));
end;
\end{lstlisting}
```

The resulting code is depicted in Listing 5.

Listing 5 Useless code.

```
for i:=maxint to 0 do
begin
    j:=square(root(i));
end;
```

Also note the `float` option that makes the listing float. Instead of the `caption` option, one might prefer the `title` option, which outputs the argument without the "Listing" label. To globally change the label name from "Listing", add to your document preamble e. g.,

```
\renewcommand{\lstlistingname}{Algorithm}
```

Please read the package documentation for more information on the `lstlisting` environment and how to adapt it locally.

The other above-mentioned recommended packages to include code or algorithms, `algorithm` and `algorithm2e`, are not preloaded. It is up to the authors to load one of these packages. If so, please use the option `ruled` to ensure the correct placement of the caption (as mentioned before, captions of text-like content are placed above the content and left aligned), for instance `\usepackage[ruled]{algorithm}` or `\usepackage[ruled]{algorithm2e}`. For further details, please read the documentation of these packages.

2.10   Bibliography and References

LIPIcs uses BibTeX to format references. Thereby the BibTeX style `plainurl` is used for BibTeX processing. Please do not change the bibliographic style. LIPIcs only allows for numerical citation and forbids author-year citations. (So the `natbib` package is not used by `lipics-v2021`.)

All entries of the bibliography have to be referenced in the text by using the `cite` macro. So do not use the command `\nocite{*}` to add unreferenced bibliographic entries to the bibliography.

In general, references should be as complete as possible. So the authors should not abbreviate author names or journal names in references and should ensure that all mandatory fields according to the BibTeX standard are available, e. g., no errors and warnings about the referenced BibTeX entries should occur. If possible references should contain an external link (e. g., DOI or URL). DOI is preferred.

For some guidelines regarding document internal references, please see Section 2.4.

Hint. The *dblp computer science bibliography*[40] indexes more than 5 million publications from major computer science outlets. A large amount of the indexed publications also come with a DOI. Furthermore, *dblp* provides a BIBTEX export service, so that exported entries can be easily reused in the paper's bibliography. It is highly recommended to use *dblp* to enrich the references and add missing DOIs, for example.

2.11 Line numbers

The `lineno` package is loaded, and line numbering is switched on by default. To (globally) switch line numbering off, use `\nolinenumbers` (in the document preamble).

`\nolinenumbers`

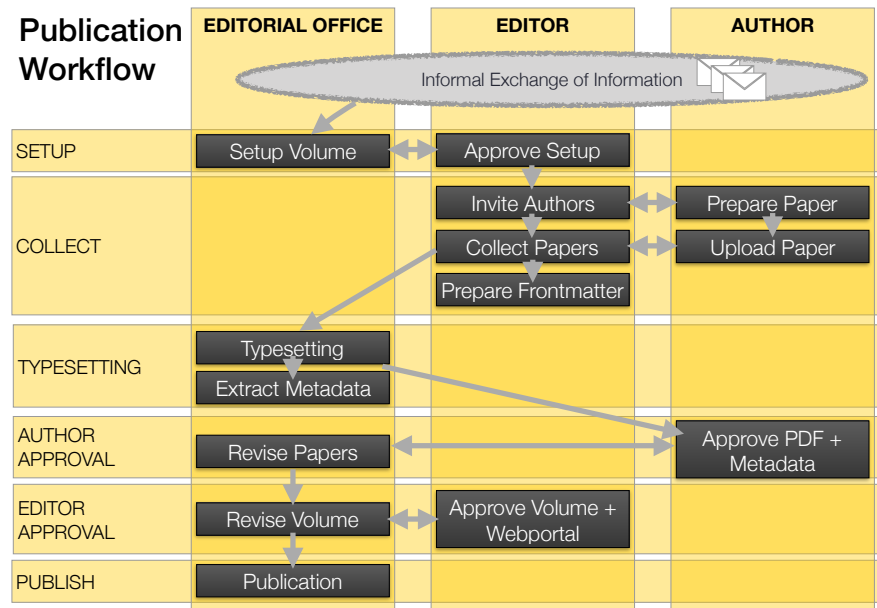## 2.12 Packaging and Submission

Please provide a self-contained archive file containing all files necessary to compile/process the document. It is mandatory that there is exactly one LATEXfile per article (and no include-files for sections/subsection) and one BIBTEX file. All other files like graphics should be part of the archive.

---

[40] `https://dblp.org`; Search: `https://dblp.org/search/`

3 Publication Workflow and Final Typesetting

After the submission of the camera-ready version of the document to the Dagstuhl Publishing submission server, the document will be checked/revised by the editors and finally by Dagstuhl Publishing.

3.1 Submission Workflow

Figure 5 Publication Workflow.

Figure 5 gives an overview of the various steps on the way to publishing a volume/article that take place after the review process. Usually 6–12 months before the planned publication of the volume, the LIPIcs office clarifies some necessary details (e.g. the aimed date of publication and a raw and preliminary schedule). This **informal exchange of information** usually happens by E-mail. The actual **publication workflow** of the volume starts approx. 12–16 weeks before the planned publication (e.g. the conference) and contains the following sub-steps, which all take place in the Dagstuhl Submission System `https://submission.dagstuhl.de`.

1. During the **setup** phase, the LIPIcs office and the editors determine the key data of a new volume: schedule, editor details, paper categories (such as "Invited Talk", "Short Paper", "Regular Paper") and the author agreement. This step ends up with a web-portal for the volume on the Dagstuhl submission server (being set-up by Dagstuhl Publishing) which guides editors and authors through all further steps.
2. Using this web-portal, the editors then **collect** the camera-ready versions of the articles to be published. They inform the authors about the acceptance of their article and invite them to submit their camera-ready version directly to the submission system.
   a. The authors **prepare** their article according to the instructions provided in this document and upload a zip-archive containing all source files as described in Section 2.12 along with a signed author agreement.

- During the submission, the LaTeX document is compiled on the server.
- Only correct and complete documents are accepted.
- After the submission of the LaTeX document, the author is asked for missing (mandatory) metadata.

b. The editors monitor the submission process and take care that all authors submit complete source files in due time. It is not expected that the editors compile/revise/typeset the provided sources.

c. After the submission, the editors check the final order of the submitted articles within the submission tool and prepare a draft of the **frontmatter**[41].

3. After the volume has been submitted to Dagstuhl Publishing by the editors, the LIPIcs office carefully checks all papers and revises them to achieve compliance with the instructions provided in this document. These checks include the actual **typesetting** of the documents as well as the respective **metadata**. All changes (and open issues, if any) are documented in a changelog provided to the authors in the next step.

4. Usually two to three weeks before publication of the volume, the LIPIcs office asks the **authors for their approval** of the revised documents and the metadata. To avoid any delay of the publication process, this author approval phase is limited to 3–5 days. In order to enable all authors to participate, the exact period will be announced to the authors at least one week in advance. Authors may then also authorize other users to perform the approval for them, if necessary. If the authors (or authorized users) nevertheless allow the approval period to expire, the corresponding article is considered passively approved.

5. After a possible revision of the documents and/or their metadata incorporating the feedback of the authors, the LIPIcs office asks the **editors for approval** of all documents and the complete volume.

6. After the editor approval, the LIPIcs office officially **publishes** the proceedings volume (including DOI registration, registration for long-term archiving, submission to indexing services like dblp or Google Scholar).

All above mentioned sub-steps of the submission process are actively supported by the Dagstuhl Submission System introduced in Spring 2019.

3.2 Final Typesetting

The final typesetting of the document is done by the LIPIcs office before asking for the authors' approval for publication. In this phase, the LIPIcs office tries to adapt/revise the documents as little as possible, while still achieving a consistent appearance in line with the corporate identity of the LIPIcs series. Basically, the articles are revised to ensure compliance with the standards described in this document. Nevertheless, the publisher tries to maintain the personal touch of the authors. If the instructions given in this document and the template file (`lipics-v2021-sample-article.tex` provided with the style archive) have been followed closely, only minimal changes will be made to the articles, if any.

The internal workflow during the typesetting process is essentially based on the list of requirements given in Section 1. In particular,

- the `\documentclass` and paper-format of the article will be checked;
- individual (non-style-compliant) spacing and font modifications will be removed;

---

[41] The editors only have to prepare the editor details, the preface, and details regarding the conference organisation. Volume-specific information and the final table of contents will be added by the LIPIcs office during the finalisation of the volume.

- the top matter and mandatory metadata fields will be checked/corrected;
- default sectioning commands will be applied wherever possible and `\texorpdfstring{...}{...}` macros will be inserted for non-ascii headlines;
- default enumeration/itemization/description styles will be applied wherever possible (see Section 2.6 and Appendix B);
- default theorem-like environments and the corresponding proof environments will be used wherever possible (see Sections 2.7 and Appendix C);
- default math environments will be applied wherever possible (see Section 2.8);
- figures, tables and listings/algorithms will be revised to fit the standards described in Section 2;
- in particular, wrapped figures will be converted into ordinary figure-environments;
- all overfull `\hbox`es will be removed;
- quotation marks will be standardized to "...", dashes will be standardized to " – ";
- in exceptional cases, font-sizes and scaling are revised to improve the readability;
- the BibTEX file is revised by adding missing mandatory fields (according to the BibTEX standard) and enabling or adding URL/DOIs where applicable.

References

1 Mark Trettin and Jürgen Fenn. An essential guide to LATEX2$_\epsilon$ usage – obsolete commands and packages, 2007. English Translation of German version. URL: `https://ctan.org/pkg/l2tabu-english`.

A   Packages used in `lipics-v2021`

Table 4 lists all packages used in `lipics-v2021`.

Table 4 Packages used in `lipics-v2021`.

Package	URL	Selected Options
microtype	https://www.ctan.org/pkg/microtype	
inputenc	https://www.ctan.org/pkg/inputenc	utf8
lmodern	https://www.ctan.org/pkg/lmodern	
fontawesome5	https://www.ctan.org/pkg/fontawesome5	
fontenc	https://www.ctan.org/pkg/fontenc	T1
textcomp	https://www.ctan.org/pkg/textcomp	
eucal	https://www.ctan.org/pkg/eucal	mathscr
amssymb	https://www.ctan.org/pkg/amssymb	
soul	https://www.ctan.org/pkg/soul	
color	https://www.ctan.org/pkg/color	
xcolor	https://www.ctan.org/pkg/xcolor	usenames
babel	https://www.ctan.org/pkg/babel	
amsmath	https://www.ctan.org/pkg/amsmath	tbtags,fleqn
enumerate	https://www.ctan.org/pkg/enumerate	
graphicx	https://www.ctan.org/pkg/graphicx	
array	https://www.ctan.org/pkg/array	
multirow	https://www.ctan.org/pkg/multirow	
tabularx	https://www.ctan.org/pkg/tabularx	
threeparttable	https://www.ctan.org/pkg/threeparttable	online
listings	https://www.ctan.org/pkg/listings	
lineno	https://www.ctan.org/pkg/lineno	left,mathlines
hyperxmp	https://www.ctan.org/pkg/hyperxmp	
totpages	https://www.ctan.org/pkg/totpages	
hyperref	https://www.ctan.org/pkg/hyperref	pdfa,unicode
colorprofiles	https://www.ctan.org/pkg/colorprofiles	
caption	https://www.ctan.org/pkg/caption	labelsep=space, singlelinecheck=false, font=up,small, labelfont=sf,bf, listof=false
rotating	https://www.ctan.org/pkg/rotating	figuresright
subcaption	https://www.ctan.org/pkg/subcaption	
comment	https://www.ctan.org/pkg/comment	
xstring	https://www.ctan.org/pkg/xstring	
amsthm	https://www.ctan.org/pkg/amsthm	
cleveref	https://www.ctan.org/pkg/cleveref	capitalise, noabbrev
aliascnt	https://www.ctan.org/pkg/aliascnt	
thmtools	https://www.ctan.org/pkg/thmtools	

B   `lipics-v2021` Styles of Lists, Enumerations, and Descriptions

List of different predefined enumeration styles:

- `\begin{itemize}...\end{itemize}`
- ...
- ...

1. `\begin{enumerate}...\end{enumerate}`
2. ...
3. ...

(a) `\begin{alphaenumerate}...\end{alphaenumerate}`
(b) ...
(c) ...

  (i) `\begin{romanenumerate}...\end{romanenumerate}`
 (ii) ...
(iii) ...

(1) `\begin{bracketenumerate}...\end{bracketenumerate}`
(2) ...
(3) ...

Description 1 `\begin{description} \item[Description 1]  ...\end{description}`
Description 2 Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
Description 3 ...

C   `lipics-v2021` Theorem-like Environments

List of different theorem-like environments:

▶ **Theorem 3.** `\begin{theorem} ...\end{theorem}`   theorem
Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

▶ **Lemma 4.** `\begin{lemma} ...\end{lemma}`   lemma
Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

▶ **Corollary 5.** `\begin{corollary} ...\end{corollary}`   corollary
Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

▶ **Proposition 6.** `\begin{proposition} ...\end{proposition}`   proposition
Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

▶ **Exercise 7.** `\begin{exercise} ...\end{exercise}`   exercise
Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.

`definition`	▶ **Definition 8.** `\begin{definition}` ...`\end{definition}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`conjecture`	▶ **Conjecture 9.** `\begin{conjecture}` ...`\end{conjecture}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`observation`	▶ **Observation 10.** `\begin{observation}` ...`\end{observation}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`example`	▶ **Example 11.** `\begin{example}` ...`\end{example}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`note`	▶ Note 12. `\begin{note}` ...`\end{note}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`note*`	▶ Note. `\begin{note*}` ...`\end{note*}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`remark`	▶ Remark 13. `\begin{remark}` ...`\end{remark}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`remark*`	▶ Remark. `\begin{remark*}` ...`\end{remark*}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`claim`	▷ Claim 14. `\begin{claim}` ...`\end{claim}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`claim*`	▷ Claim. `\begin{claim*}` ...`\end{claim*}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
`proof`	**Proof.** `\begin{proof}` ...`\end{proof}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque. ◀
`claimproof`	Proof. `\begin{claimproof}` ...`\end{claimproof}`
	Fusce eu leo nisi. Cras eget orci neque, eleifend dapibus felis. Duis et leo dui. Nam vulputate, velit et laoreet porttitor, quam arcu facilisis dui, sed malesuada risus massa sit amet neque.
	◁