

Рекомендації щодо оформлення тез

Оргкомітет конференції

Інститут прикладної математики та фундаментальних наук
Національного університету "Львівська політехніка"

Файл повинен починатися рядками

```
\documentclass[twoside,11pt]{article}
\usepackage{psc_13}
\begin{document}
```

Команда `\usepackage` підключає стильовий файл `psc_13.sty`.

Якщо тези пишуться англійською мовою, то це треба вказати явно, відразу після `\begin{document}` (інакше не будуть коректно розставлятись англійські переноси слів):

```
\English
```

Далі формується заголовок тез, УДК, перелік авторів з вказуванням наукових ступенів, звань та посад, місця роботи авторів:

```
\title{Назва праці}
\udk{530.145}
\author{Автор~I.\,Б., Співавтор1~I.\,Б., Співавтор2~I.\,Б.}
[\textbf{Автор~I.\,Б.}$^{1,2}$, д.ф.-м.н., проф. каф.\,ВМ
\newline
\textbf{Співавтор1~I.\,Б.}$^1$, к.т.н., доц. каф.\,ВМ
\newline
\textbf{Співавтор2~I.\,Б.}$^2$, к.ф.-м.н., доц. каф.\,ВМ]
\organization
{\address{1}Національний університет "Львівська політехніка"\newline
(вул.\,С.\,Бандери,\,12, м.\,Львів, 79013, Україна)
\address{2}Інститут фізики конденсованих систем НАН України \newline
(вул.\,Свенціцького,\,1, м.\,Львів, 79011, Україна)}
```

Якщо всі автори працюють в одній установі, то перелік авторів та назву установи слід оформити так:

```
\title{Назва праці}
\udk{530.145}
\author{Автор~I.\,Б., Співавтор1~I.\,Б., Співавтор2~I.\,Б.}
[\textbf{Автор~I.\,Б.}, д.ф.-м.н., проф. каф.\,ВМ
\newline
\textbf{Співавтор1~I.\,Б.}, к.т.н., доц. каф.\,ВМ
\newline
\textbf{Співавтор2~I.\,Б.}, к.ф.-м.н., доц. каф.\,ВМ]
\organization
{Національний університет "Львівська політехніка"\newline
(вул.\,С.\,Бандери,\,12, м.\,Львів, 79013, Україна)}
```

Ці команди можна задавати в довільному порядку і вони повинні завершуватися командою:

```
\maketitle
```

Текст можна розбивати на розділи та параграфи командами

```
\section{Назва розділу}
\paragraph{Назва параграфа.}
```

Команди `\subsection`, `\subparagraph` не передбачені, оскільки тези мають невеликий обсяг. У кінці назв розділів крапка не ставиться. Назва параграфу є частиною першого рядка абзацу; якщо ця назва є повним реченням, то крапка ставиться перед закриваючою фігурною дужкою.

Тези повинні закінчуватись командою

```
\end{document}
```

Усі тези в збірнику починаються з нової сторінки, що дозволяє зберігати задане автором розташування матеріалу на сторінках. Прохання до авторів – не використовувати команди скорочення вертикальних проміжків та інші способи штучного ущільнення тексту.

Обсяг тез – одна або дві повні сторінки. Для оформлення тез зручно використати файл-зразок `example_13.tex`. Після редагування цього файлу його потрібно перейменувати на прізвище першого автора латиницею (наприклад, `Shevchenko.tex`).

Стандартні засоби L^AT_EX'у

Немає особливих обмежень на використання основних засобів L^AT_EX'у [?, 2–4]. В тези можна вставляти формули, таблиці, списки, рисунки, виноски і т. д.

Стандартні пакети, які підключаються в стильовому файлі `psc_13.sty`: `algorithm`, `algorithmic`, `amssymb`, `amsmath`, `array`, `babel`, `balance`, `color`, `epic`, `euscript`, `graphicx`, `ifthen`, `inputenc`, `mathrsfs`, `pb-diagram`, `theorem`, `url`, `xu`. Цими пакетами можна користуватися, не викликаючи команду `\usepackage`. Бажано обмежитись використанням лише цими пакетами.

Формули всередині тексту, навіть дуже короткі, слід оточувати знаками долара `$`:

число <code>-\$-3{,}14\$</code>	число <code>-3,14</code> – правильно
число <code>-3,14</code>	число <code>-3,14</code> – неправильно
змінна <code>~\$x\$</code>	змінна <code>x</code> – правильно
змінна <code>x</code>	змінна <code>x</code> – неправильно

Виключні формули без номера оточуються дужками `\[` та `\]`. Виключні формули з номером оточуються командами `\begin{equation}` і `\end{equation}`. Команда `\label{name}` між ними задає мітку формули. Кириличні букви в іменах міток `name` не допускаються. Мітка дозволяє посилатись на формулу командою `\eqref{name}`, наприклад, команда `\eqref{eqcases}` дає (1).

Списки оформляються стандартними оточеннями `enumerate` або `itemize`. В стилі `psc_13.sty` визначене оточення `enumerate*` для списків, у яких, згідно із правилами української пунктуації:

- 1) номери відділяються дужкою;
- 2) пункти починаються з малої букви;
- 3) і закінчуються крапкою з комою.

Цей список зручний для перерахування коротких пунктів, що вміщаються в один рядок. Якщо пункти довші, то краще скористатись стандартним оточенням `enumerate`, вказавши після `\begin{enumerate}` команду `\afterlabel`, яка перевизначить крапку після номера на дужку.

Таблиці створюються оточенням `tabular` і оформляються як плаваючі об'єкти за допомогою оточення `table`. Підпис створюється *над таблицею* командою `\caption`, див. таблицю 1. Команда `\label`, що визначає посилання на номер таблиці, повинна йти після `\caption`.

Якщо таблиця не вміщається по ширині тексту, то можна зменшити шрифт до `\small` або навіть `\footnotesize`, чи зменшити інтервали між колонками: `\tabcolsep=2pt`.

Таблиця 1. Підпис розміщується над таблицею.

Завдання	CCEL	boosting
Cancer	3.46 ± 0.37 (3.16)	4.14 ± 1.48
German	25.78 ± 0.65 (1.74)	29.48 ± 0.93
Hepatitis	18.38 ± 1.43 (2.87)	19.90 ± 1.80

Ілюстрації готуються у форматі EPS. Для перетворення файлів формату PNG або JPEG в EPS використовуйте утиліту `bmeps`, яка входить в пакет `MiKTeX`.

Ілюстрації у растрових форматах (`png`, `jpeg`) повинні мати роздільну здатність не менше 600 dpi.

Не забудьте надіслати графічні файли разом з `TeX`-файлом! Імена графічних файлів повинні починатися з прізвища першого автора латиницею, як і основний файл тез.

Бажано, щоб ілюстрації були чорно-білими або `grayscale` (відтінки сірого). При чорно-білому друці передачу кольорів важко передбачити. Рисунки вставляються командою `\includegraphics`. Якщо рисунок займає по висоті більше за 1-2 см, то він оформляється як плаваюча ілюстрація `figure`. Підпис формується під рисунком командою `\caption`, див. рис. 1

Якщо тези набираються англійською мовою, то перед `\begin{figure}` слід вставити команду `\renewcommand{\figurename}{Fig}`, яка змінить слово “Рис” на “Fig”.

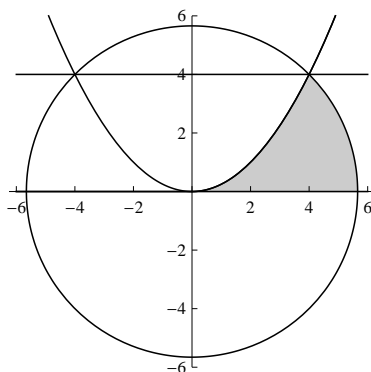


Рис. 1. Підпис повинен розміщатися під малюнком.

Виноски створюються командою `\footnote{text}`¹.

Список літератури формується оточенням `thebibliography`. Кожен запис бібліографії починається командою `\bibitem{name}`. Мітка `name` дозволяє посилатися на даний запис командою `\cite{name}`. В посиланнях дозволяється вказувати кілька міток через кому: `\cite{name1,name2}`. Нова команда `\citenb` дає посилання без квадратних дужок, що дозволяє створювати діапазони. Наприклад, `[?–4]` було отримано так: `[\citenb{Voronlatex}--\citenb{Lvovsky}]`. Кириличні букви в іменах міток `name` використовувати не дозволяється. Записи сортуються за порядком посилання на них.

Прізвища та ініціали авторів виділяються командою `\Bibauthor`. Назви статей у збірниках виділяються командою `\Bibtitle`. Якщо публікація існує тільки в електронному вигляді, то веб-посилання створюється командою `\Biburl`.

Посилання на сайти формуються командою `\url`. При створенні документа в форматі `pdf` посилання стають активними, хоча не підкреслюються і не виділяються кольором. Приклад: `\url{http://lp.edu.ua}`.

¹Текст виноски вказується в аргументі `text`.

Математичні позначення

Дотримання наведених нижче рекомендацій сприяє більшій однотипності в позначеннях та полегшує підготовку збірника тез.

Як десятковий розділювач використовуємо кому: у формулі $\{, \}14$, у тексті 3,14.

Числові множини \mathbb{N} , \mathbb{Z} , \mathbb{R} , \mathbb{C} задаємо командами `\NN`, `\ZZ`, `\RR`, `\CC`.

В стилі `psc_13.sty` переозначені команди `\geq`, `\leq`, `\emptyset`, `\epsilon`, `\kappa`, `\phi` математичних символів \geq , \leq , \emptyset , ε , κ , φ .

Математичні оператори `lim`, `inf`, `sup`, `min`, `max` переозначені так, що межі завжди ставляться знизу, а не збоку.

Визначені математичні оператори: `argmin`, `argmax`, `diag`, `sign`, `tr`, `const` командами `\argmin`, `\argmax`, `\diag`, `\sign`, `\Tr`, `\const`.

Команди `\myop` и `\mylim` створюють нові оператори, не передбачені \LaTeX 'ом:

<code>\myop{Ker} f</code>	$\text{Ker } f$
<code>A_{\myop{Ker} f}</code>	$A_{\text{Ker } f}$
<code>\myop{Hom}_{\Phi}(A,B)</code>	$\text{Hom}_{\Phi}(A, B)$
<code>\mylim{Hom}_{\Phi}(A,B)</code>	$\underset{\Phi}{\text{Hom}}(A, B)$

Для виділення векторних і матричних величин прямим жирним шрифтом передбачена команда `\vec{формула}`.

Лінійна алгебра:

<code>\rank A</code>	$\text{rank } A$
<code>\Tr A</code>	$\text{tr } A$
<code>\diag (d_1, \dots, d_n)</code>	$\text{diag}(d_1, \dots, d_n)$
<code>A^T</code>	A^T
<code>u^T F^T F u</code>	$u^T F^T F u$
<code>\vec{x}</code>	\mathbf{x}
<code>\Omega \neq \vec{\Omega}</code>	$\Omega \neq \boldsymbol{\Omega}$
<code>e^{-\vec{x}^T \Sigma x}</code>	$e^{-\mathbf{x}^T \Sigma \mathbf{x}}$ (правильно)
<code>e^{-x^T \Sigma x}</code>	$e^{-x^T \Sigma x}$ (неправильно)

Теорія ймовірностей:

<code>\Prob{x \colon x \in A}</code>	$P\{x: x \in A\}$
<code>\Expect \xi</code>	$E\xi$
<code>\Var \xi</code>	$D\xi$
<code>\Normal(\mu, \Sigma)</code>	$\mathcal{N}(\mu, \Sigma)$
<code>p(x \cond y)</code>	$p(x y)$

В умовних ймовірностях команда `\cond` задає правильні пробіли навколо вертикальної риси.

Для оформлення умовних конструкцій використовуйте стандартне оточення `cases`. Текст всередині формул виводиться командою `\text`:

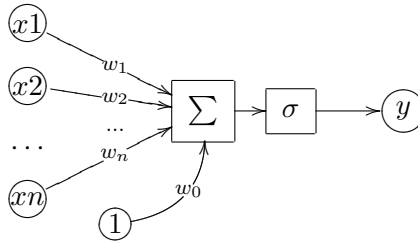
$$y(x, \alpha) = \begin{cases} -1, & \text{якщо } f(x, \alpha) < 0; \\ +1, & \text{якщо } f(x, \alpha) \geq 0. \end{cases} \quad (1)$$

```
\begin{equation}\label{eqcases}
y(x, \alpha) = \begin{cases}
-1, & \text{\text{якщо } } f(x, \alpha) < 0; \\
+1, & \text{\text{якщо } } f(x, \alpha) \geq 0.
\end{cases}
\end{equation}
```

Щоб розмір дужок відповідав розміру формули, використовують команди `\left` и `\right`. Проте в простих випадках ці команди лише захаращують текст. Краще записати `f(x_i)` замість `f\left(x_i\right)` – результат в обох випадках буде однаковий.

Для вставки матриці в рядок тексту ($\begin{pmatrix} a & b & c \\ 1 & 2 & 3 \end{pmatrix}$) використовуйте оточення `smallmatrix`. Усі інші способи дають некрасивий результат.

Малювання графів за допомогою оточення `network` з пакета `Xy-pic`. В стилі `psc_13.sty` визначено дві допоміжні команди. Команда `\nnNode` задає ім'я та координати вершини, а команда `\nnLink` зв'язує дві раніше іменовані вершини. Зовнішній вигляд вершин та зв'язків задається засобами пакета `Xy-pic`:



Команди, за допомогою яких був намальований цей персептрон:

```
\begin{network}
  \nnNode"x1"(0,7)      {+[o][F]{x1}}
  \nnNode"x2"(0,2)      {+[o][F]{x2}}
  \nnNode"dd"(0,-3)     {{{\cdots}}}
  \nnNode"xn"(0,-7)     {+[o][F]{xn}}
  \nnNode"1"(7,-9)      {+[o][F]{1}}
  \nnNode"sum"(14,0)    {++[F-]{\sum}}
  \nnNode"sig"(21,0)    {++[F-]{\sigma}}
  \nnNode"y"(30,0)      {++[o][F]{y}}
  \nnLink"x1,sum"       {@{->}|{w_1}}
  \nnLink"x2,sum"       {@{->}|{w_2}}
  \nnLink"dd,sum"       {@{}|{\dots}}
  \nnLink"xn,sum"       {@{->}|{w_n}}
  \nnLink"1,sum"        {@{->}@/_3ex/|{w_0}}
  \nnLink"sum,sig"      {@{->}}
  \nnLink"sig,y"        {@{->}}
\end{network}
```

Алгоритми оформляються в стилі псевдокоду за допомогою плаваючого оточення `Algorithm`. Усередині оточення `Algorithm` стандартні ключові слова `\IF`, `\FOR`, `\WHILE`, та ін., дають, відповідно, **якщо**, **для**, **поки**, і т.д. Кроки алгоритму нумеруються автоматично, тому на них можна посилатися, див. важливий крок 5 алгоритму 1.

Оточення типу теорем. Наступні оточення виводять текст *похилим шрифтом*: `def` або `definition` – Означення, `theorem` – Теорема, `lemma` – Лема, `state` – Твердження, `corollary` – Наслідок.

Приклад використання:

```
\begin{Theorem}
Формулювання теореми.
\end{Theorem}
```

У результаті отримуємо:

Теорема 1. *Формулювання теореми.*

Алгоритм 1. Показані всі команди керування.

Вхід: x, y ;

Вихід: $z = F(x, y)$;

- 1: ініціалізація: $b := a$;
 - 2: для $i = 1, \dots, n$
 - 3: для всіх $w \in W$ таких, що $w \geq 0$
 - 4: повторювати
 - 5: важливий крок: обчислення вектора u_i ;
 - 6: поки $\|u_i - u_{i-1}\| > \varepsilon$;
 - 7: якщо $a > 0$ то
 - 8: поки $W \neq \emptyset$
 - 9: $W := W - \{a\}$;
 - 10: інакше якщо $a = 0$ то
 - 11: цикл // нескінченний цикл
 - 12: за певних умов вихід;
 - 13: інакше // при $a < 0$
 - 14: $a := 1$;
-

```
\begin{Algorithm}[h]
\caption{Показані всі команди керування.}
\label{Algeexample}
  \REQUIRE  $x, y$ ;
  \ENSURE  $z = F(x, y)$ ;
  \STATE ініціалізація:  $b := a$ ;
  \FOR{\$i=1, \dots, n\$}
    \FORALL{\$w \in W\$ таких, що  $w \geq 0$ }
      \REPEAT
        \STATE важливий крок: обчислення вектора  $u_i$ ; \label{algcalscu}
        \UNTIL{\$ \|u_i - u_{i-1}\| > \epsilon \$};
      \ENDFOR
    \ENDFOR
  \IF{\$a > 0\$}
    \WHILE{\$W \neq \varnothing\$}
      \STATE  $W := W - \{a\}$ ;
    \ENDWHILE
  \ELSIF{\$a = 0\$}
    \LOOP[нескінченний цикл]
      \STATE за певних умов \EXIT;
    \ENDLOOP
  \ELSE[ при  $a < 0$ ]
    \STATE  $a := 1$ ;
  \ENDIF
\end{Algorithm}
```

Оточення **axiom** — Аксіома, **problem** — Задача, **example** — Приклад, **remark** — Зауваження, **hypothesis** — Гіпотеза виводять текст звичайним шрифтом.

Якщо тези набирають англійською мовою, то перед оточенням **theorem** слід поставити `\renewcommand{\theoremname}{Theorem}`, що призведе до заміни слова “Теорема” на “Theorem”. За необхідності аналогічні переозначення потрібно зробити і для інших оточень:

```
\renewcommand{\lemmaname}{Lemma}
\renewcommand{\statename}{State}
\renewcommand{\corollaryname}{Corollary}
\renewcommand{\defname}{Definition}
```

```
\renewcommand{\definitionname}{Definition}
\renewcommand{\Proofname}{Proof}
\renewcommand{\axiomname}{Axiom}
```

Крім того, визначено оточення `myEnvironment`, в якому передбачено задання автором назви оточення.

Приклад використання:

```
\begin{myEnvironment}{\bf Нове оточення}
\sl Оточення використовують, якщо стандартні не влаштовують.
\end{myEnvironment}
```

У результаті отримуємо:

Нове оточення. *Оточення використовують, якщо стандартні не влаштовують.*

Рекомендації з оформлення

Деякі правила типографіки. Дужки усіх видів набираються впритул до тексту, який вони оточують. Розділові знаки набираються впритул до попереднього тексту і окремо від наступного.

Лапки в українському тексті створюються так: «*текст*», а в англійському – так: ‘*text*’.

Використовувати символ " не можна!

Три крапки в тексті та у формулах задаються командою `\dots`.

Тире відділяється від попереднього тексту нерозривним пробілом:

Знання~---_сила

В довгих словах з дефісом, наприклад, «диференціально-символьний», дефіс ставиться командою `"=`, інакше слово не буде переноситись:

диференціально"=символьний

Команда `"~` забороняє перенос на дефісі: *F*-перетворення, $\$F\$$ "~перетворення.

Нерозривний пробіл `~` ставиться між коротким прийменником та наступним словом, а також між дуже короткою формулою та зв'язаним з нею за змістом словом: число~ $\$N\$$ в~ $\$k\$$ разів більше за~ $\$n\$$.

Між формулами, що розміщені поруч, іноді потрібний додатковий пробіл:

$\$a=1,b=2\$$		$a = 1, b = 2$	— погано
$\$a=1\$,\ \$b=2\$$		$a = 1, b = 2$	— краще
$\$a=1\$,\ \backslash:\ \$b=2\$$		$a = 1, b = 2$	— добре
$\$a=1\$,\ \backslash;\ \$b=2\$$		$a = 1, b = 2$	— добре

Іноді в формулі виникає потреба забрати пробіли навколо знака операції. Наприклад, якщо знак \times використовується не як добуток, а для вказування розмірів матриці або растрового зображення, то його краще не оточувати пробілами:

$\$640\backslash\times 480\$$		640×480	— погано
$\$640\{\backslash\times\}480\$$		640×480	— добре

Додатковий пробіл `\quad` рекомендується вставляти між довгими виразами, які розміщені через кому в виключній формулі.

Короткий пробіл `\,` ставиться після знака номера: `\No\,6`; в ініціалах: Я.\,В.\,Антонов; в скороченнях: і.\,т.\,д.; і.\,т.\,п.

Не слід використовувати жирний шрифт для виділення *важливих слів* або *термінів*. Це робиться командою `\emph{текст}`.

Правила форматування вихідного коду полегшують його читання і роботу над ним:

- починайте кожне речення з нового рядка;
- набирайте окремим рядком команди `\begin`, `\end`, `$$`, `\[`, `\]`, `\section`, `\subsection`, `\paragraph`, `\item`, `\bibitem`, `\par`, `\label`;
- внутрітекстові формули, крім зовсім коротких, набирайте окремим рядком;
- довгі описи формул розбивайте на рядки; використовуйте табуляції для виділення вкладених дужок і логічно відокремлених частин формул, як показано в прикладі 1.

Приклад 1. Форматування складної формули:

$$R'_N(F) = \frac{1}{N} \sum_{i=1}^N N \left(P(+1 | x_i) C(+1, F(x_i)) + P(-1 | x_i) C(-1, F(x_i)) \right).$$

```
\begin{align*}
R'_N(F)
= \frac{1}{N} \sum_{i=1}^N
\Bigl(
& P(+1 \cond x_i) C\bigl(+1, F(x_i)\bigr)
+{} \ \ \ {}+{}
& P(-1 \cond x_i) C\bigl(-1, F(x_i)\bigr)
\Bigr).
\end{align*}
```

- [1] *Воронцов К. В.* $\LaTeX 2_{\epsilon}$ в примерах. — 2006. — <http://www.ccas.ru/voron/latex.html>.
- [2] *Гуссенс М., Миттельбах Ф., Самарин А.* Путеводитель по пакету \LaTeX и его расширению $\LaTeX 2_{\epsilon}$. — Москва: Мир, 1999. — 606 с.
- [3] *Котельников И. А., Чеботаев П. З.* $\LaTeX 2_{\epsilon}$ по-русски. — Новосибирск: Сибирский хронограф, 2004. — 489 с.
- [4] *Львовский С. М.* Набор и вёрстка в пакете \LaTeX . — 3-е издание. — Москва: МЦНМО, 2003. — 448 с.