Documentation for package ahsan

The LaTeX packages can be found at https://github.com/AnglyPascal/sty/. This package requires the memoir documentclass. To use this package you need to add the following in the preamble.

```
\usepackage[<class>, <theme>, <other features>]{ahsan}
```

For example:

```
\usepackage[book, darkblue, nosecthm, noindex]{ahsan}
```

Fonts

The following fonts are used in this package: CMU Serif, CMU Sans Serif, Share, Share Tech Mono, Titillium Web, Andika New Basic, Yanone Kaffeesatz

Because of the Type 1 fonts used, either XeLaTeX or LuaLaTeX is needed to compile the document. Alternatively, the nofont option can be passed to stop loading manual fonts.

The default font family is Sans Serif. To set Serif as the family default, pass the option serif. Sans fonts can be used in math mode too. To use sans in math mode, pass the option mathsans.

Classes

There are currently four classes in this package: book, article, bangla, beamer.

The book option basically toggles on every fancy customization like fancy theorems, header-footers, fonts and indices. It uses the chapter/section/subsection division, which is perfect for using in larger projects.

The article option doesn't switch on fancy header-footer and font customization by default, and is ideal for articles or handouts. It is in the developmental phase, as using it alongside memoir doesn't allow using section/subsection as the default division, and requires chapters.

The bangla option is also in the developmental phase. It currently toggles the packages polyglossia and fontspec to provide support for basic editing in Bengali. But it is not fully functional, and the page numbers and other titles will still be in English. In the future, bookstyle and latexbangla packages will be merged with it to provide full Bengali support.

The beamer option is for writing beamer presentations. There are two themes inspired from Nord color palettes, NordDark, NordWhite.

The comparision between the default settings of the book and the article class is given below. Note that these settings can be switched on off with addition options.

settings	default	book	article
fancy headers	false	true	false
theorem definitions	true	true	true
fancy theorems	false	true	true
section theorem numbering	false	true	false
indices	false	true	false
tableofconent depth of 4	false	true	false
figures	true	true	true
other settings	false	true	false
font customization	true	true	true
default font family	sans	sans	sans

Indices

There are 5 different indices, `prob`, `thm`, `def`, `strat` and the default one. You can add indices manually by passing the option `index` and stop adding them by passing the option `nointex`.

A demostration for their usage:

```
\index[prob]{A problem}
\index[thm]{Number Theory!A theorem under number theory}
\index[strat]{Greedy Algorithm!A problem that's solved by being greedy}
\index[def]{Combinatorics!Graph Theory!Bipartite Graph}
\index{There is a fun story here}

\printindex[prob]
\printindex
```

The index stlye is in `indStyle.ist`, that has to be in the main directory to have effect.

```
headings_flag 1

heading_prefix "\n\\large\\sffamily%
\\noindent\\texttt{"heading_suffix "}%
    \\par\\nopagebreak\n"

item_0 "\n \\item \\small "

delim_0 " \\hfill "
delim_1 " \\dotfill "
delim_2 " \\dotfill "
```

Themes

There are currently 5 fully functioning themes available in this package. They are `darkblue`, `darkpink`, `monocolor`, `lightgreen`, and the default theme. Not mentioning any specific theme in the options will toogle the default theme.

The definitions of the colors for these themes are in `mycolors.sty` file. This package and can be imported as a standalone package using

```
\usepackage[<theme name>]{mycolors}
```

The default theme. It uses a bluish color palette in a clear white background. classical.

Theorem 1 (Title) — This is a generic theorem in generic colors.

Theorem 2 (Title) — An important theorem's box.

Proof. And here is a proof that proves something.

Definition (Name)— Some definition

Problem 1 (Title). A trivial problem, that is already solved

Solution. The solution to the problem goes here, maybe we can prove it with a lemma

Lemma— A lemma

Proof. the proof of the lemma

This is the **darkBlue** theme, suitable for the dark theme users. It focuses on a blue color palette.

Theorem 1 (Title) — This is a theorem in dark blue theme.

Theorem 2 (Title) — An important theorem's box.

Proof. And here is a proof that proves something.

Definition (Name)— Some definition

Problem 1 (Title). And here is a problem with lights off

Solution. The solution to the problem goes here, maybe we can prove it with a lemma

Lemma— A lemma

Proof. the proof of the lemma

This is the **monocolor** theme, suitable for B&W printing. It extends the gray palette to fill in all the niches.

Theorem 1 (Title) — This is a theorem in black and white colors

Theorem 2 (Title) — An important theorem's box.

Proof. And here is a proof that proves something.

Definition (Name)— Some definition

Problem 1 (`Title`). A trivial problem, that is already solved

Solution. The solution to the problem goes here, maybe we can prove it with a lemma

Lemma— A lemma

Proof. the proof of the lemma

This is the **darkPink** theme, suitable for the dark theme users. It focuses on a pink color palette.

Theorem 1 (Title) — This is cute theorem in pink

Theorem 2 (Title) — An important theorem's box.

Proof. And here is a proof that proves something.

Definition (Name)— Some definition

Problem 1 (`Title`). A trivial and cute problem, that is already solved

Solution. The solution to the problem goes here, maybe we can prove it with a lemma

Lemma— A lemma

Proof. the proof of the lemma

This is the **lightGreen** theme, for the nature lovers. It uses a light green color palette.

Theorem 1 (Title) — This is a theorem in the forests.

Theorem 2 (Title) — An important theorem's box.

Proof. And here is a proof that proves something.

Definition (Name)— Some definition

Problem 1 (`Title`). A trivial problem, that is already solved

Solution. The solution to the problem goes here, maybe we can prove it with a lemma

Lemma— A lemma

Proof. the proof of the lemma

The themes are implemented using global names for various environments, and then specifying those names for each color. For example, the default theme is defined by

```
\definecolor{NavyBlue}{HTML}{05445e}
\definecolor{NordBlack}{HTML}{3B4252}
\definecolor{BlueGrotto}{HTML}{189ab4}
\definecolor{NordWhite}{HTML}{E5E9F0}
\definecolor{BabyBlue}{HTML}{d4f1f4}
\definecolor{NordGreen}{HTML}{A3BE8C}
\definecolor{Teal}{HTML}{29a0b1}
\definecolor{TealGreen}{HTML}{167d7f}

\colorlet{pageC}{white}                          % page color
\colorlet{fgC}{black}                            % text color

\colorlet{thmC}{NavyBlue}                        % theorem color
\colorlet{probC}{NordBlack}                      % problem color
\colorlet{defC}{BlueGrotto}                      % definition color
\colorlet{solC}{fgC}                             % solution color

\colorlet{thmBgC}{NordWhite!95!BabyBlue}         % theorem background color
\colorlet{probBgC}{pageC}                        % problem background color
\colorlet{defBgC}{NordWhite!95!NordGreen}        % definition background color

\colorlet{urlC}{TealGreen}                       % url color
\colorlet{linkC}{Teal}                           % link color
\colorlet{fileC}{TealGreen}                      % internal file color
\colorlet{impC}{NordBrightBlue!80!fgC}           % used in important parts
```

Theorem Styles

The package allows six options for theorem customization.

`thm`: the default, allows the usage of many theorem environments
`nothm`: only allows the basic theorem, problem, lemma and proof environments
`secthm`: theorem numbering follows section numbering
`nosecthm`: theorems follow their own numbering
`fancythm`: theorem environments are defined using `tcolorbox`
`nofancythm`: theorems are defined using `thmtools`

These are included in the `tcbs.sty` package which is then imported inside `ahsan.sty` for better readability.

A snippet of the implementation is followed:

```
% define the theorem style
\declaretheoremstyle[
    headfont=\bfseries\color{thmC},
    headpunct={\color{thmC}\textbf{---}}, % thmC is given by mycolors package
    postheadspace={0pt},
    spaceabove=20pt,
    spacebelow=15pt,
    notebraces={(}{)},
    notefont=\color{thmC}\mdseries\normalfont,
    bodyfont=\Titillium\selectfont,
]{theoremstyle}

\ifAhsanfancythm % if fancy theorem is on
    \ifAhsansecthm % numberwithin section
        \declaretheorem[style=theoremstyle, name=Theorem, numberwithin=section]
        {theorem}
    \else
        \declaretheorem[style=theoremstyle, name=Theorem]{theorem}
    \fi

    % put the theorem inside a tcolorbox
    \tcolorboxenvironment{theorem}{
        blanker,breakable,left=1em,colback=thmBgC,colframe=thmBgC,coltext=fgC,
        before skip=25pt, after skip=20pt, borderline west={.15em}{0pt}{thmC}
    }
\else
    % define generic theorems
    \ifAhsansecthm
        \newtheorem{theorem}{Theorem}[section]
    \else
        \newtheorem{theorem}{Theorem}
    \fi
\fi
```

Environments that are included in the package and are accessed by allowing fancy theorems:

```
theorem     lemmma            proposition  corollary
problem     generalization    definition
solution    prooof            % yes, there is an extra 'o'
algo        data_structure    game         construction
sollem      soldef            % to use lemmas and definitions inside a solution
claim       remark            note         case
fact        conjecture        answer       joke
exercise    example           question
take_note   % textbox to put interesting ideas
myitemize   % itemize with fancy borders
```

Macros

To provide ease of use, the package comes with mane pre-defined macros. The environment related macros can be found as part of the `tcbs.sty` file, and other macros are located in the `macros.sty` file.

If fancy theorems are allowed, then useful environment related macros including the following onese are defined.

```
\theo{<theorem link>}{<theorem name>}{<theorem statement>}
\lem{<lemma link>}{<lemma name>}{<lemma statement>}
\den{<definition link>}{<definition name>}{<definition statement>}

\thmbox{<theorem link>}{<theorem name>}{<theorem>} % boxed theorem
\impden{<definition link>}{<definition name>}{<definition>} % boxed definitions

\prob{<problem link>}{<problem name>}{<difficulty>}{<problem statement>}
\gene{<generalization link>}{<name>}{<details>}

\prop{<proposition name>}{<proposition>}
\coro{<corollary>}
\algorithm{<algo name>}{<link to the source>}{<algo details>}
\dstruct{<data structure name>}{<link to the source>}{<ds details>}

\proof{<proof>}
\solu{<solution>}
\nt{<note>}

\impeq{<math equation>}
```

Implementations of some of these macros are quite tricky. Because a simple problem or theorem macro is doing the following things

1. Adds a hyperref in the title, if no link or problem name is not given, then don't add this part.

2. Adds the problem statement
3. Labels the problem by its name. If problem name not given, don't add a label.
4. Adds the problem name to the index. If name not given, add it under `Unknown Source` in the indices.

A sample implementation for the `prob` macro is given below:

```
\newcommand{\prob}[4]{
    \ifthenelse{\isempty{#1}}{% link not given, don't call hyperref
        \ifthenelse{\isempty{#2}}{% name not given, name it unknown source
            \begin{problem}
                \index[prob]{unknown source}
                #4
            \end{problem}
        }{%
            \begin{problem}[#2]
                \index[prob]{#2}    % adds it to the problem index
                \label{problem:#2}  % labels the problem by its name
                #4
            \end{problem}
        }%
    }{% if both informations are given, add the hyperref
        \begin{problem}[\href{#1}{#2}]
            \index[prob]{#2}
            \label{problem:#2}
            #4
        \end{problem}
    }%
}
```

There are two macros for including png and pdf files, respectively `fig` and `figdf`. The usage of the macros are given below:

```
\fig{<scale>}{<png filename>}{<caption>}
\figdf{<scale>}{<pdf filename>}{<caption>}

% <scale> is a number between 0 to 1, the relative width lenght, for example:
\figdf{.5}{picture}{a picture with a width of half the text width}
% will add the picture.pdf at the center with width .5\textwidth

% the defintion of the figure is given by
\newcommand{\figdf}[3]{
    \begin{figure}[H]
        \centering
        \includegraphics[width=#1\textwidth]{#2.pdf}
        \caption{#3}
    \end{figure}
}
```

Bengali Support

Currently this package doesn't have full Bengali support. In the future there will be updates to this part. But for the time being, it is advided to use the **template** at

https://github.com/AnglyPascal/book_template

This is a fork of the template by my senior, Adib Hasan. This template has some new fancier text boxes and some fixes related to figure and page numbering, and also supports more essential macros. Overall this template uses a monocolor based theme for black and white publication.

To use the packages one needs to add the following in their preamble:

```
1  \usepackage{bookstyle_exp}
```