WinEdt and TeX: step-by-step guide...

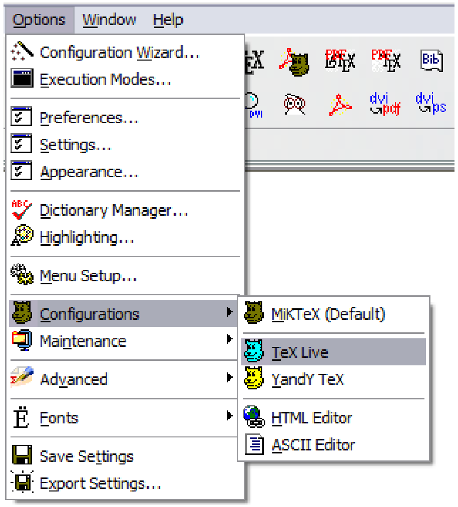
WinEdt is by default *already* configured for MiKTeX. No changes to PATH are required in order to make WinEdt work with properly installed MiKTeX. All you have to do after installing or updating a TeX System is to restart WinEdt in order to give it a chance to detect your TeX System or Ghostscript. Reinstalling WinEdt is *not* required and will not improve the integration with other accessories. Below is an illustrated tutorial explaining the idea of writing LaTeX documents with WinEdt and MiKTeX. The instructions are written with a first-time user in mind, although more experienced users, too, might benefit from reading and learning about WinEdt's capabilities, which have been designed to assist you maintain and navigate your source code...

In this section we show WinEdt in action by illustrating the *Edit-Compile-Preview* cycle. Note that WinEdt's documentation does not deal with the syntax of LaTeX. The fastest way to learn [La]TeX is to base your work on the examples of similar nature. Extensive documentation and literature about the art of typesetting in TeX is also widely available. You'll find downloads, links and references on TeX Users Group (TUG) home page:

http://www.tug.org

Here we focus on WinEdt's role in TeX-ing! Not all TeX Systems include TeXify accessories and not all previewers support src specials and DVI Search. Our example works with MiKTeX 2.4 (default WinEdt's Configuration) which supports many extras (beyond basic TeX-ing)...

WinEdt's `Options -> Configurations` Menu allows you to instantly switch between MiKTeX and TeX Live configurations should you decide to give both TeX Systems a try:

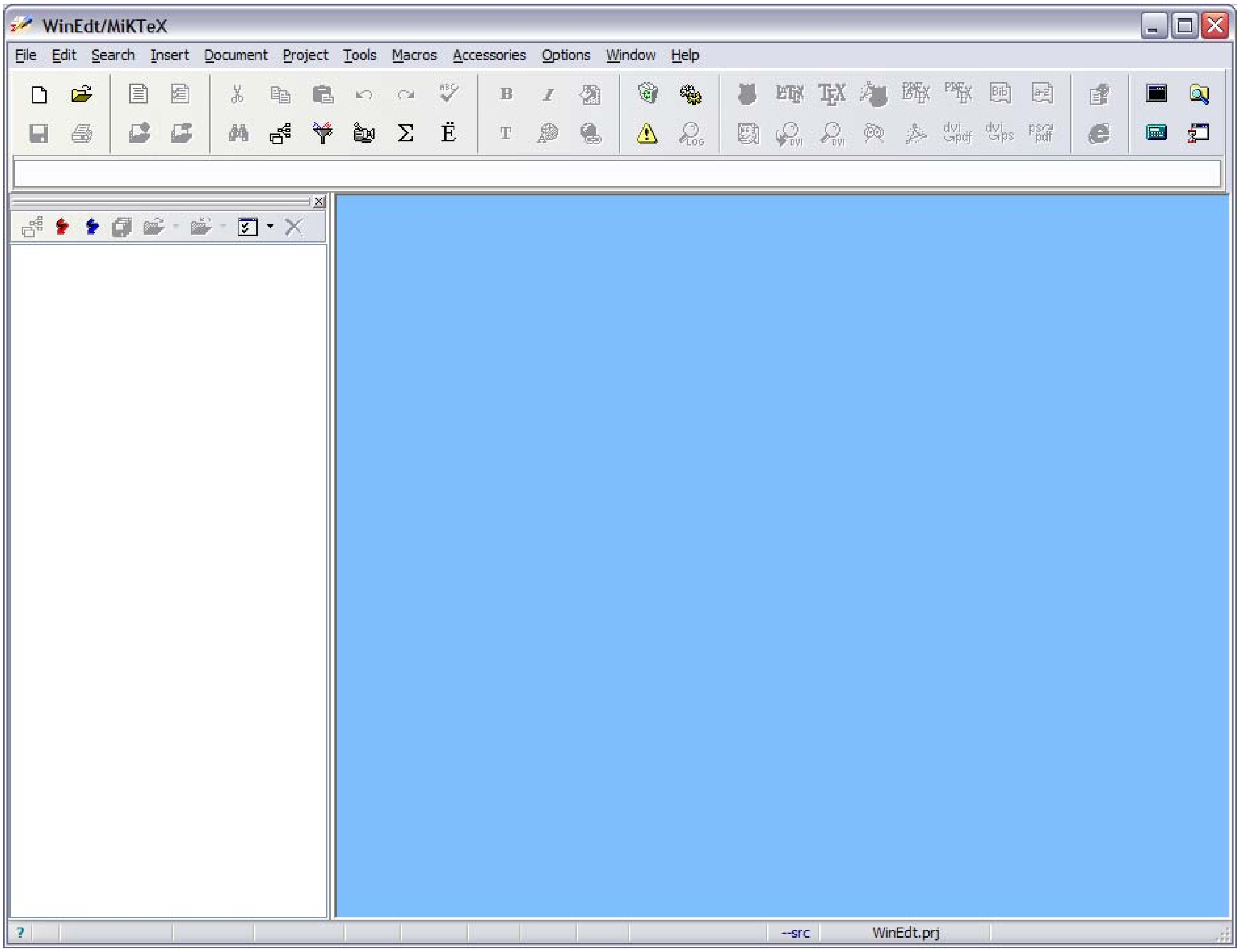


If you want to use WinEdt with TeX Live, you should execute the menu command:

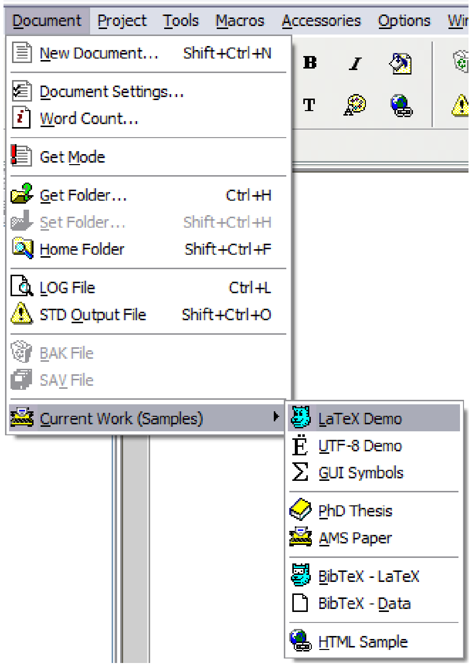
```
Options -> Configurations -> TeX Live
```

This will reconfigure WinEdt for *TeX Live* distribution. After that WinEdt will automatically detect *properly* installed *TeX Live* System and the rest of the instructions in this section apply.

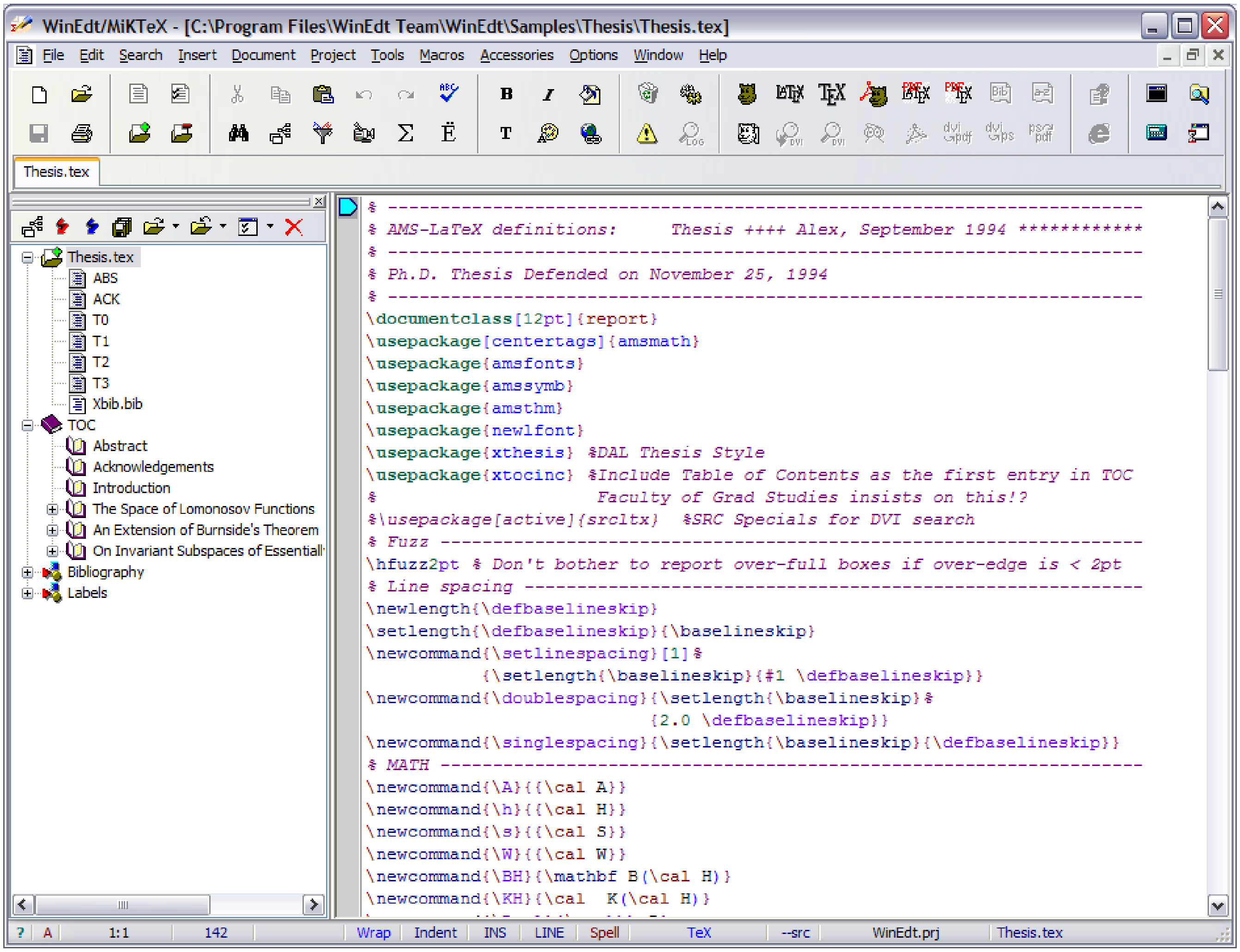
An "empty" copy of WinEdt is running:


WinEdt/MiKTeX
File Edit Search Insert Document Project Tools Macros Accessories Options Window Help
--src
WinEdt.prj

Most of the Menu Items and associated tool bar buttons are disabled. In particular, all TeX Icons are currently disabled. This does not mean that WinEdt cannot find your TeX. It simply indicates that currently there is no TeX document loaded. As soon as you'll open or create a TeX document the compilation commands will become enabled. For our purpose let's open a Sample Thesis that comes with WinEdt:



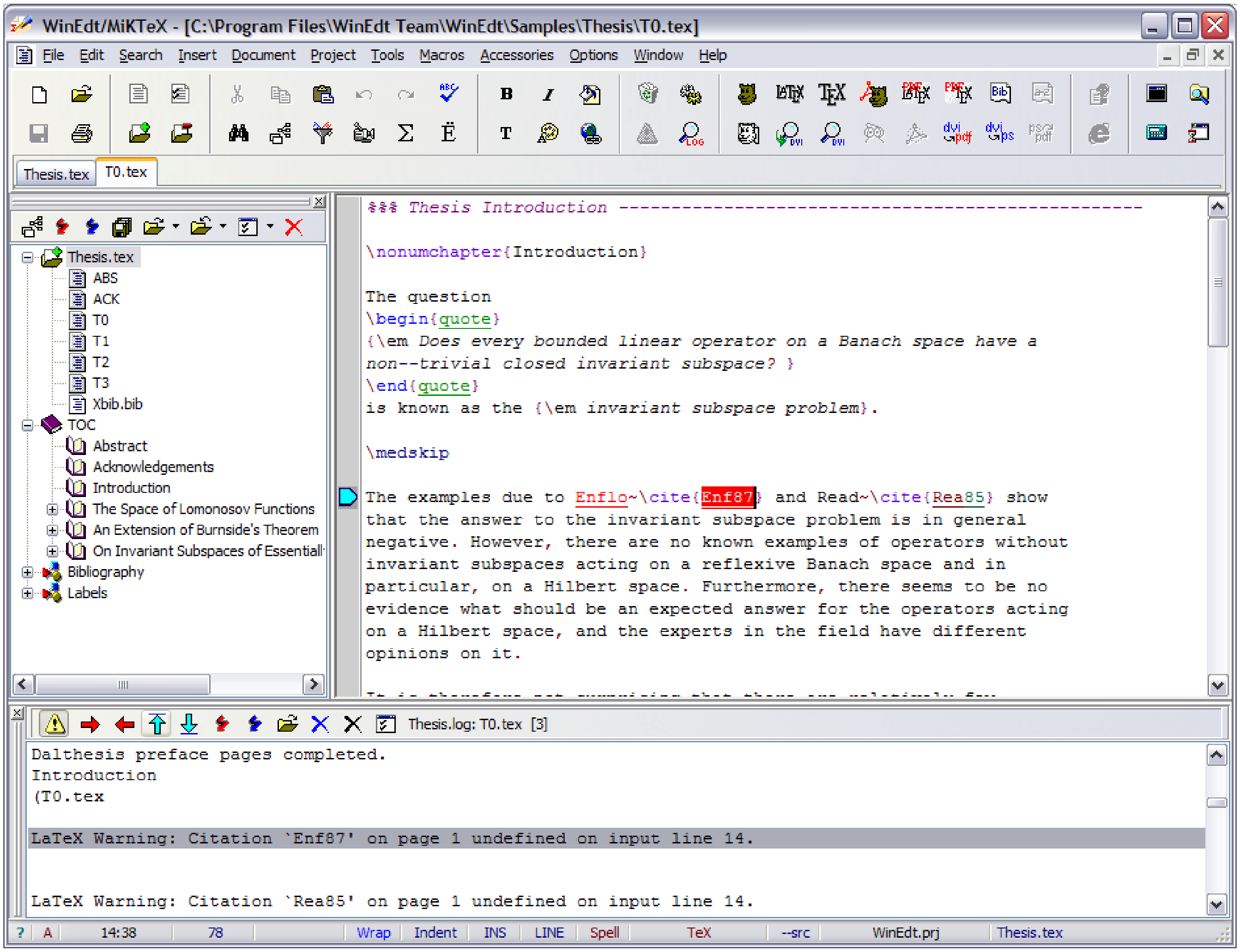
After the file is opened, build the tree and navigation bar by pressing the first icon in the tree's control toolbar (docked in on left). That's more like it:



```
% ------------------------------------------------------------------------
% AMS-LaTeX definitions:     Thesis ++++ Alex, September 1994 ************
% ------------------------------------------------------------------------
% Ph.D. Thesis Defended on November 25, 1994
% ------------------------------------------------------------------------
\documentclass[12pt]{report}
\usepackage[centertags]{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{amsthm}
\usepackage{newlfont}
\usepackage{xthesis} %DAL Thesis Style
\usepackage{xtocinc} %Include Table of Contents as the first entry in TOC
%                     Faculty of Grad Studies insists on this!?
%\usepackage[active]{srcltx}  %SRC Specials for DVI search
% Fuzz -------------------------------------------------------------------
\hfuzz2pt % Don't bother to report over-full boxes if over-edge is < 2pt
% Line spacing -----------------------------------------------------------
\newlength{\defbaselineskip}
\setlength{\defbaselineskip}{\baselineskip}
\newcommand{\setlinespacing}[1]%
           {\setlength{\baselineskip}{#1 \defbaselineskip}}
\newcommand{\doublespacing}{\setlength{\baselineskip}%
                           {2.0 \defbaselineskip}}
\newcommand{\singlespacing}{\setlength{\baselineskip}{\defbaselineskip}}
% MATH -------------------------------------------------------------------
\newcommand{\A}{{\cal A}}
\newcommand{\h}{{\cal H}}
\newcommand{\s}{{\cal S}}
\newcommand{\W}{{\cal W}}
\newcommand{\BH}{\mathbf B(\cal H)}
\newcommand{\KH}{\cal  K(\cal H)}
```

Compilation commands such as [PDF] TeXify, LaTeX and TeX are now enabled indicating that we can compile the current TeX Project with the compiler of our choice (since this is a LaTeX document we have to use LaTeX version of the TeX compiler). MiKTeX's TeXify accessory can be used to perform a full compile (run the compiler, BibTeX and MakeIndex as many times as necessary to update the references, bibliography and index). This would be the fastest way to compile the document but for educational purposes we'll take care of things manually. Note that certain commands are still disabled. There is good reason for that: DVI Preview or dvi2ps require the dvi file (which will be created by the TeX Compiler) and currently this file doesn't exists. Let's make it! Ready? Press the LaTeX button...

By default (depending on your choice of the Execution Mode) the Console Window (with black background) appears and LaTeX is doing its thing: complaining about some undefined references and citations and displaying the page numbers as it progresses through the text. When there are no Errors the Windows disappears quite fast and you are not expected to process the feedback from the console Window. In WinEdt you can Press the `Ctrl+E` (Errors... Command in the Search Menu) shortcut to display the LOG file at the bottom of the screen. This file contains all the information produced by TeX. Furthermore, subsequent command search the LOG file for errors and warnings and display the corresponding line in WinEdt. Although the warnings in the current LOG file are not serious (they will disappear once we compile the bibliography and re-compile the source) let's try it (you can either keep on pressing Ctrl+E to "walk" through the errors or else use the GUI in the Error Form). To close the Errors dialog click on the X in the left top corner of the form or use the Shift+Ctrl+E shortcuts....


WinEdt/MiKTeX - [C:\Program Files\WinEdt Team\WinEdt\Samples\Thesis\T0.tex]
File
Edit
Search
Insert
Document
Project
Tools
Macros
Accessories
Options
Window
Help
Thesis.tex
T0.tex
Thesis.tex
ABS
ACK
T0
T1
T2
T3
Xbib.bib
TOC
Abstract
Acknowledgements
Introduction
The Space of Lomonosov Functions
An Extension of Burnside's Theorem
On Invariant Subspaces of Essentiall
Bibliography
Labels
Thesis.log: T0.tex [3]
? A 14:38 78 Wrap Indent INS LINE Spell TeX --src WinEdt.prj Thesis.tex

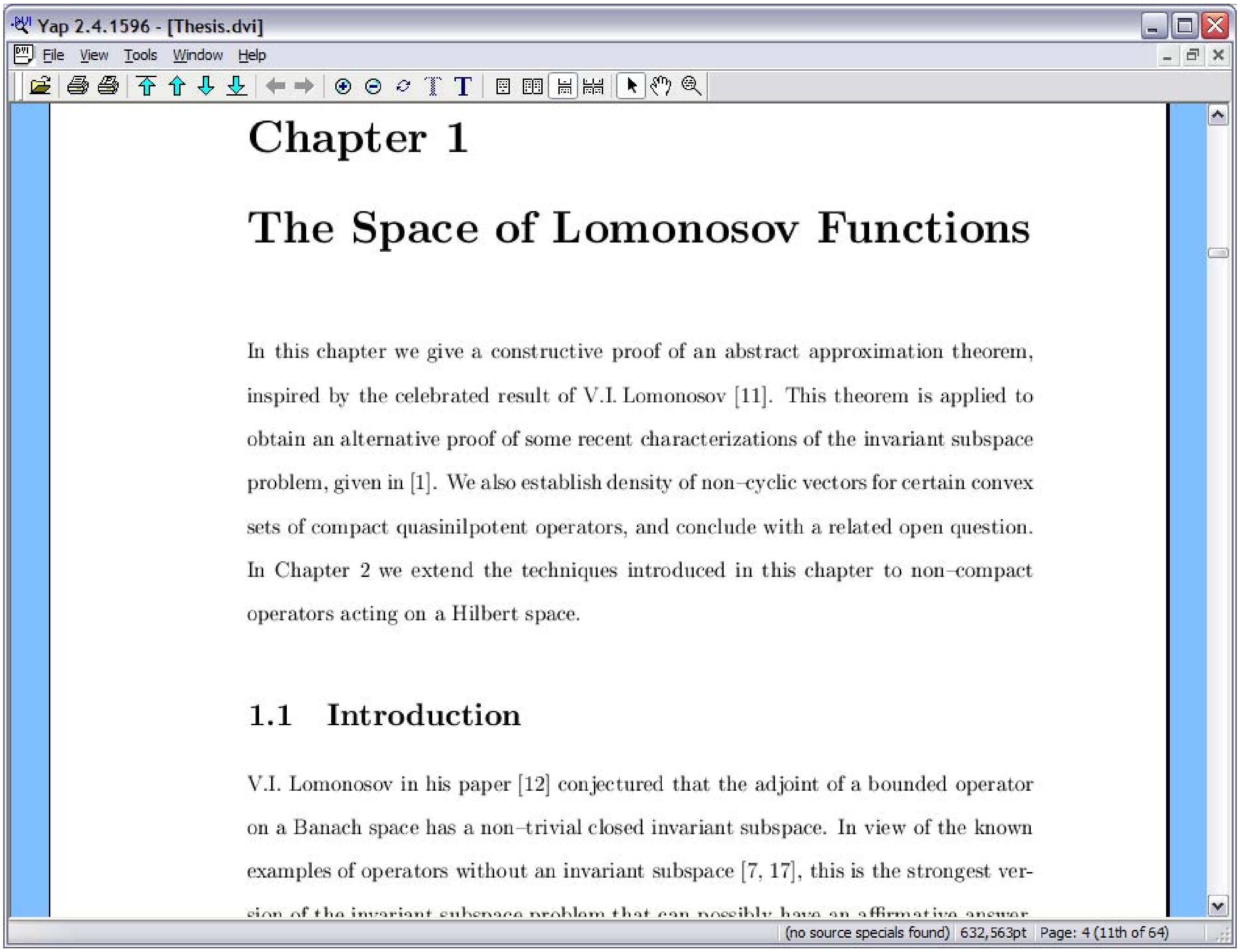
```
%%% Thesis Introduction -----------------------------------------------------

\nonumchapter{Introduction}

The question
\begin{quote}
{\em Does every bounded linear operator on a Banach space have a
non--trivial closed invariant subspace? }
\end{quote}
is known as the {\em invariant subspace problem}.

\medskip

The examples due to Enflo~\cite{Enf87} and Read~\cite{Rea85} show
that the answer to the invariant subspace problem is in general
negative. However, there are no known examples of operators without
invariant subspaces acting on a reflexive Banach space and in
particular, on a Hilbert space. Furthermore, there seems to be no
evidence what should be an expected answer for the operators acting
on a Hilbert space, and the experts in the field have different
opinions on it.
```

```
Dalthesis preface pages completed.
Introduction
(T0.tex

LaTeX Warning: Citation `Enf87' on page 1 undefined on input line 14.


LaTeX Warning: Citation `Rea85' on page 1 undefined on input line 14.
```

By the way, the location of the Error Form can be changed in the Appearance Dialog. By default the from is docked at the bottom of the Client Window but this can be changed if you don't like it there.

WinEdt and YAP

You have noticed that DVI Search and Preview and dvi2ps commands are now enabled: the file Thesis.dvi exists and the commands now make sense. Let us first get rid of all the Warnings by properly compiling the bibliography. Press the Button BibTeX and then LaTeX two more times to get all the cross-references right (TeXify does all that in one step)... There are no more Warnings or Errors. Close the LOG file (X in the top left corner of the Error Form). Now press the DVI Preview Button. The previewer starts (in MiKTeX it is called YAP). Move a few pages down and see how the compiled document looks.



Chapter 1

The Space of Lomonosov Functions

In this chapter we give a constructive proof of an abstract approximation theorem, inspired by the celebrated result of V.I. Lomonosov [11]. This theorem is applied to obtain an alternative proof of some recent characterizations of the invariant subspace problem, given in [1]. We also establish density of non–cyclic vectors for certain convex sets of compact quasinilpotent operators, and conclude with a related open question. In Chapter 2 we extend the techniques introduced in this chapter to non–compact operators acting on a Hilbert space.

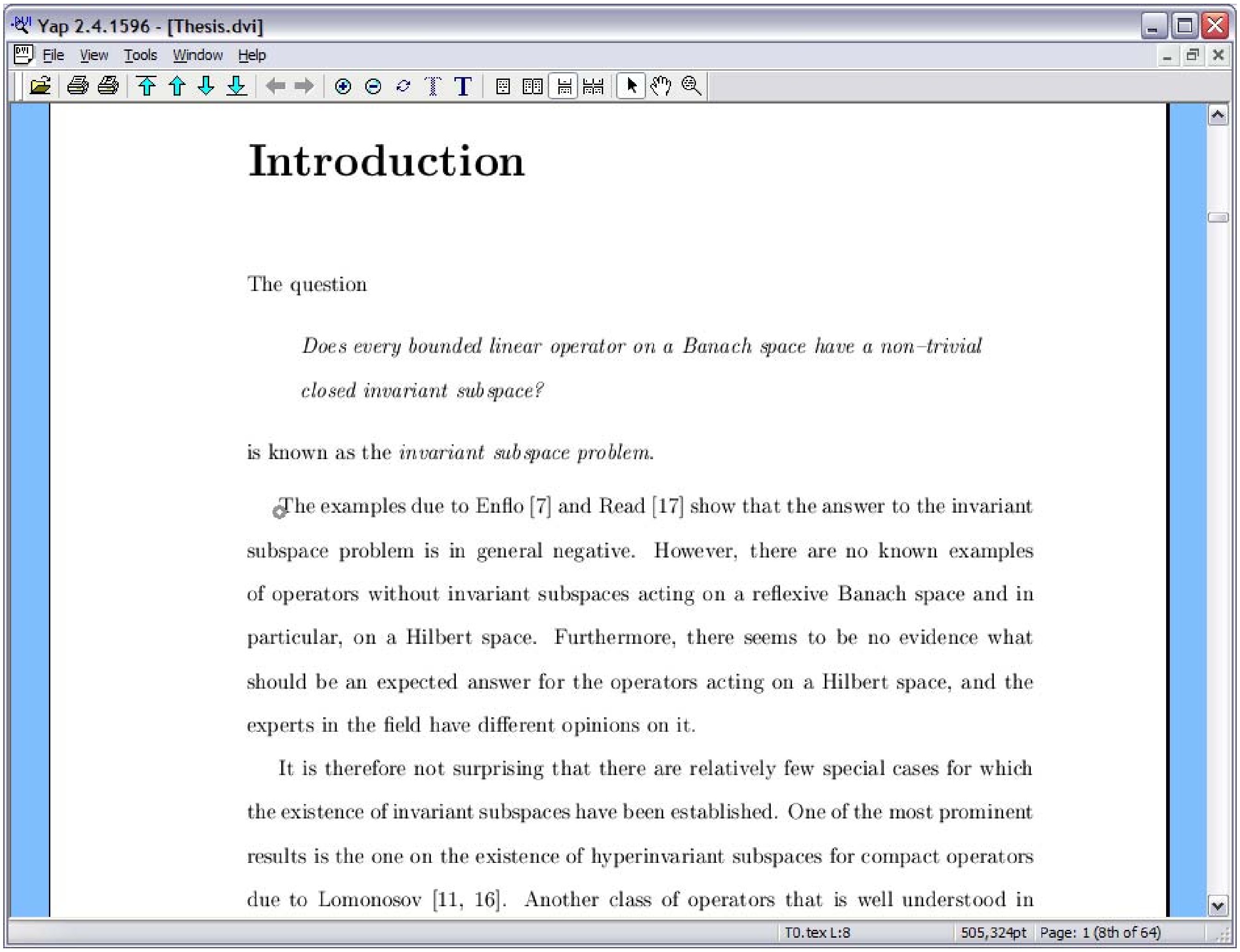
1.1 Introduction

V.I. Lomonosov in his paper [12] conjectured that the adjoint of a bounded operator on a Banach space has a non–trivial closed invariant subspace. In view of the known examples of operators without an invariant subspace [7, 17], this is the strongest version of the invariant subspace problem that can possibly have an affirmative answer

Note that your actual position was in the Introduction Chapter (`T0.tex`). So, how did WinEdt know to compile the main file `Thesis.tex` rather than the current chapter (which would fail to compile without a proper preamble present in the main file). This is because `Thesis.tex` is set as the main file. To remove or set the current file as the main document use the commands in the Project Menu or the "+/-" buttons in the toolbar. For your Convenience the name of the currently set main file is displayed in the last panel of WinEdt's Status Line. When you'll start working on your projects you should (re)set the main file to your main document or else remove it if you are working on a simple one source file TeX document.

SRC Specials and DVI Search

WinEdt's Status Line displays a panel "`--src`" (it can be toggled between empty and "`--src`" by clicking on it. The "`--src`" command qualifier instructs TeX to insert "SRC Specials" in the resulting dvi file. These specials allow the previewer to link the compiled document with the actual position in the source file (within a few lines). This makes the Forward and Inverse DVI Search possible. Go back to WinEdt: you are currently at the position of previously undefined citation. Press the Button DVI Search. YAP now starts displaying the position roughly corresponding to your current position in WinEdt. Note a circle in YAP marking the current line:



Introduction

The question

Does every bounded linear operator on a Banach space have a non–trivial closed invariant subspace?

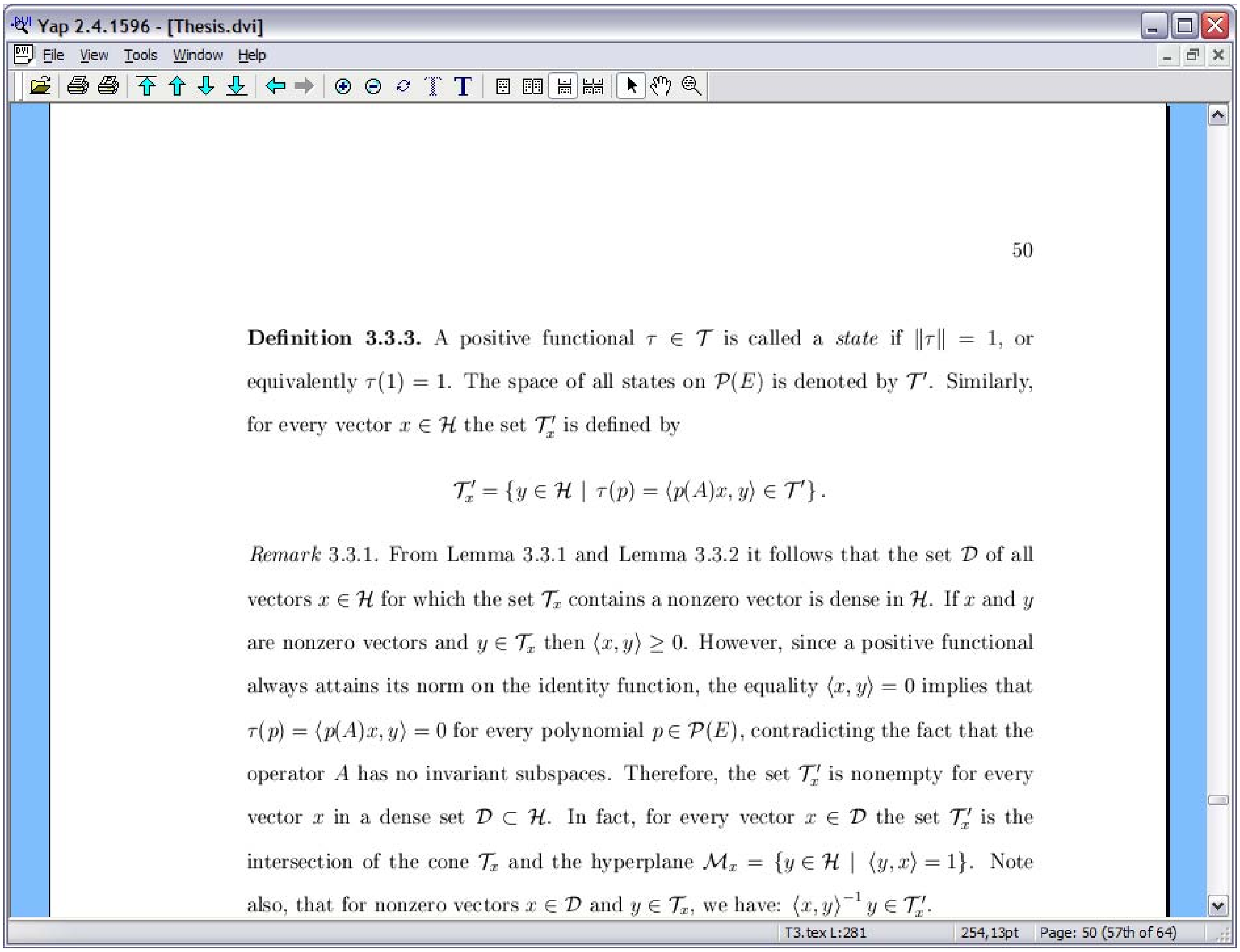
is known as the *invariant subspace problem*.

The examples due to Enflo [7] and Read [17] show that the answer to the invariant subspace problem is in general negative. However, there are no known examples of operators without invariant subspaces acting on a reflexive Banach space and in particular, on a Hilbert space. Furthermore, there seems to be no evidence what should be an expected answer for the operators acting on a Hilbert space, and the experts in the field have different opinions on it.

It is therefore not surprising that there are relatively few special cases for which the existence of invariant subspaces have been established. One of the most prominent results is the one on the existence of hyperinvariant subspaces for compact operators due to Lomonosov [11, 16]. Another class of operators that is well understood in

Inverse Search

Even more useful is the Inverse DVI Search which can be initiated from the previewer (by double-clicking on a preview) and as a result the editor displays the corresponding source line (again there is margin of error since the exact position may not be available from the inserted src specials). For example, in YAP go to Page 50 (see the `View -> Go To` Menu in YAP) and double-click on the displayed equation (at the and of the Definition 3.3.3):



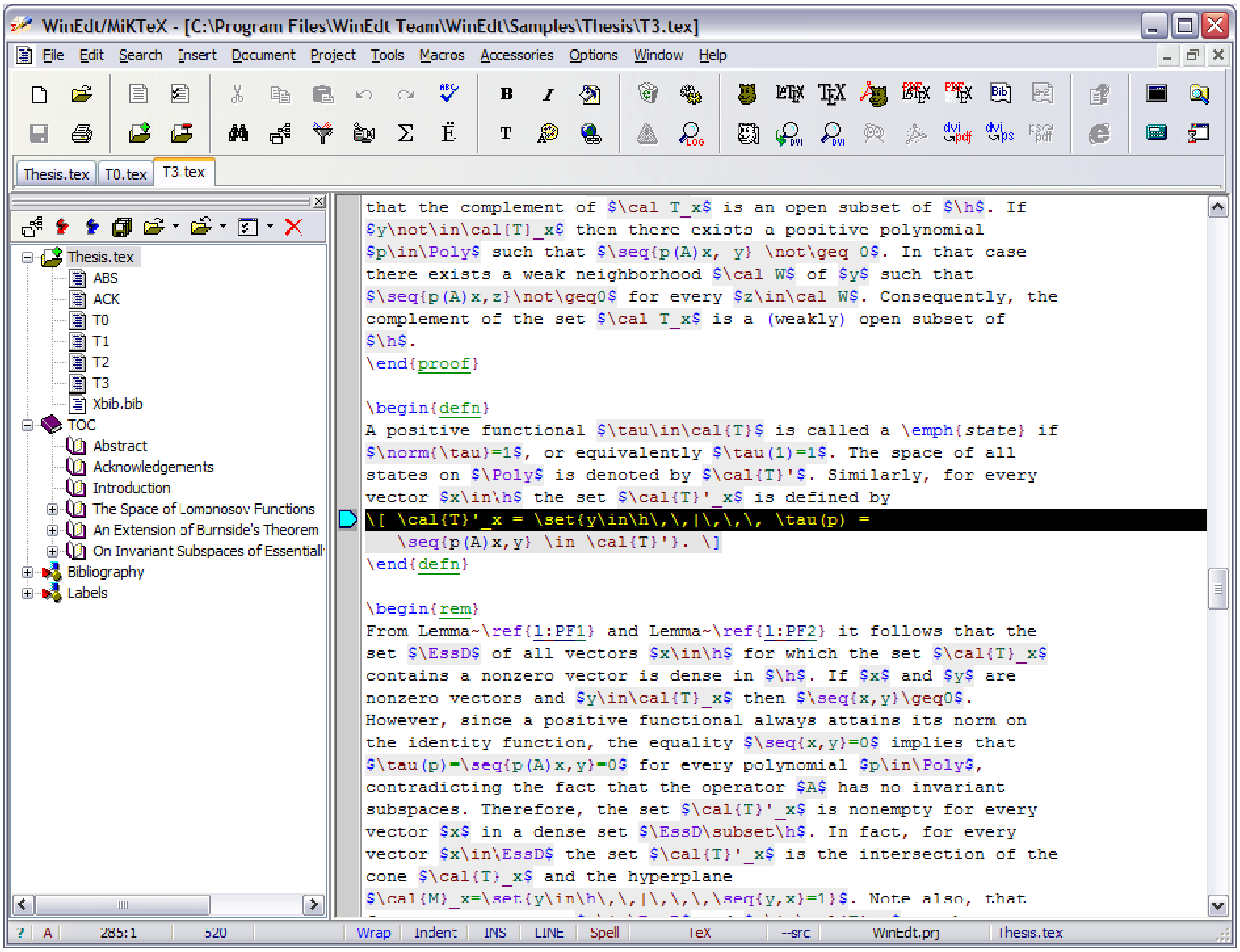
50

Definition 3.3.3. A positive functional $\tau \in \mathcal{T}$ is called a *state* if $\|\tau\| = 1$, or equivalently $\tau(1) = 1$. The space of all states on $\mathcal{P}(E)$ is denoted by $\mathcal{T}'$. Similarly, for every vector $x \in \mathcal{H}$ the set $\mathcal{T}'_x$ is defined by

$$\mathcal{T}'_x = \{y \in \mathcal{H} \mid \tau(p) = \langle p(A)x, y\rangle \in \mathcal{T}'\}.$$

Remark 3.3.1. From Lemma 3.3.1 and Lemma 3.3.2 it follows that the set $\mathcal{D}$ of all vectors $x \in \mathcal{H}$ for which the set $\mathcal{T}_x$ contains a nonzero vector is dense in $\mathcal{H}$. If x and y are nonzero vectors and $y \in \mathcal{T}_x$ then $\langle x, y\rangle \geq 0$. However, since a positive functional always attains its norm on the identity function, the equality $\langle x, y\rangle = 0$ implies that $\tau(p) = \langle p(A)x, y\rangle = 0$ for every polynomial $p \in \mathcal{P}(E)$, contradicting the fact that the operator A has no invariant subspaces. Therefore, the set $\mathcal{T}'_x$ is nonempty for every vector x in a dense set $\mathcal{D} \subset \mathcal{H}$. In fact, for every vector $x \in \mathcal{D}$ the set $\mathcal{T}'_x$ is the intersection of the cone $\mathcal{T}_x$ and the hyperplane $\mathcal{M}_x = \{y \in \mathcal{H} \mid \langle y, x\rangle = 1\}$. Note also, that for nonzero vectors $x \in \mathcal{D}$ and $y \in \mathcal{T}_x$, we have: $\langle x, y\rangle^{-1} y \in \mathcal{T}'_x$.

WinEdt gets in the foreground, Opens Chapter 3 and highlights the corresponding line:

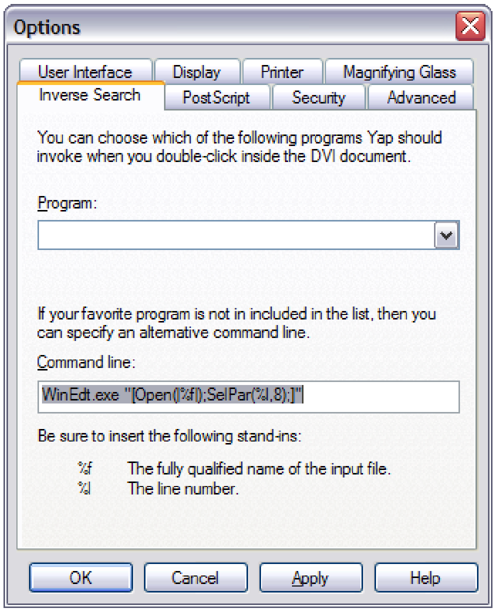


WinEdt/MiKTeX - [C:\Program Files\WinEdt Team\WinEdt\Samples\Thesis\T3.tex]
File Edit Search Insert Document Project Tools Macros Accessories Options Window Help
Thesis.tex T0.tex T3.tex
Thesis.tex
ABS
ACK
T0
T1
T2
T3
Xbib.bib
TOC
Abstract
Acknowledgements
Introduction
The Space of Lomonosov Functions
An Extension of Burnside's Theorem
On Invariant Subspaces of Essentiall
Bibliography
Labels
? A 285:1 520 Wrap Indent INS LINE Spell TeX --src WinEdt.prj Thesis.tex

```
that the complement of $\cal T_x$ is an open subset of $\h$. If
$y\not\in\cal{T}_x$ then there exists a positive polynomial
$p\in\Poly$ such that $\seq{p(A)x, y} \not\geq 0$. In that case
there exists a weak neighborhood $\cal W$ of $y$ such that
$\seq{p(A)x,z}\not\geq0$ for every $z\in\cal W$. Consequently, the
complement of the set $\cal T_x$ is a (weakly) open subset of
$\h$.
\end{proof}

\begin{defn}
A positive functional $\tau\in\cal{T}$ is called a \emph{state} if
$\norm{\tau}=1$, or equivalently $\tau(1)=1$. The space of all
states on $\Poly$ is denoted by $\cal{T}'$. Similarly, for every
vector $x\in\h$ the set $\cal{T}'_x$ is defined by
\[ \cal{T}'_x = \set{y\in\h\,\,|\,\,\, \tau(p) =
   \seq{p(A)x,y} \in \cal{T}'}. \]
\end{defn}

\begin{rem}
From Lemma~\ref{l:PF1} and Lemma~\ref{l:PF2} it follows that the
set $\EssD$ of all vectors $x\in\h$ for which the set $\cal{T}_x$
contains a nonzero vector is dense in $\h$. If $x$ and $y$ are
nonzero vectors and $y\in\cal{T}_x$ then $\seq{x,y}\geq0$.
However, since a positive functional always attains its norm on
the identity function, the equality $\seq{x,y}=0$ implies that
$\tau(p)=\seq{p(A)x,y}=0$ for every polynomial $p\in\Poly$,
contradicting the fact that the operator $A$ has no invariant
subspaces. Therefore, the set $\cal{T}'_x$ is nonempty for every
vector $x$ in a dense set $\EssD\subset\h$. In fact, for every
vector $x\in\EssD$ the set $\cal{T}'_x$ is the intersection of the
cone $\cal{T}_x$ and the hyperplane
$\cal{M}_x=\set{y\in\h\,\,|\,\,\,\seq{y,x}=1}$. Note also, that
```

DVI Search is an extremely useful feature since it allows you to quickly correct errors spotted while you are examining the compiled document. However, for this to work as indicated above you have to define the command that should be used by YAP to initiate the Inverse Search. By default, YAP may be configured so as to call some other editor (eg. Windows Notepad).

In YAP's View Menu start the Options Dialog and click on the Inverse Search Tab:



You can choose which of the following programs Yap should invoke when you double-click inside the DVI document.

Program:

If your favorite program is not in included in the list, then you can specify an alternative command line.

Command line:

```
WinEdt.exe "[Open(|%f|);SelPar(%l,8);]"
```

Be sure to insert the following stand-ins:

%f The fully qualified name of the input file.
%l The line number.

Enter

```
        WinEdt.exe "[Open(|%f|);SelPar(%l,8)]"
```

as a command line and leave the drop-down (Program) list empty. Your version of YAP may be configured differently. You should consult YAP's documentation for detailed instructions (WinEdt is irrelevant to this YAP's configuration issue). Based on the information provided by YAP (or some other previewer) you should find the way to specify the editor and enter the command line as indicated above. Not all dvi viewers support SRC specials!

You have seen enough to be able to do the same procedure with PDF LaTeX and create a pdf file which can be previewed in Acrobat Reader. Note that there is no "--src" support for this format. Alternatively, dvi2pdf converter yields a very similar result.

You can also create a `ps` file using dvi2ps converter. The resulting "ps" (postscript) file can be previewed and printed with GSView (you'll find a link on TeX Users Group home page).

Taking advantage of WinEdt's capabilities...

By now you are probably eager to start working on your own projects. However, let us use this opportunity to illustrate some advanced WinEdt features which will make your work easier...

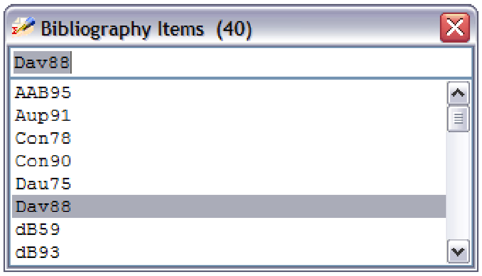
When you setup your project it is important that you set the main file and build the project tree. The items in the Project Menu (or the associated tool bar buttons) can be used to accomplish the task. Even if you don't want tree visible (in your left panel, you should let WinEdt build it for the purpose of automatic referencing).

WinEdt builds the project tree based on the Input Directives as specified in the `Options -> Advanced -> Tree` (`Tree.ini` file). Directives are mode-sensitive (since different rules apply to different document formats). By default you'll find something like:

```
[INPUT_DIRECTIVES]

//  BEGINNING_OF_LINE_ONLY
//    0 - any position is admissible
//    1 - Beginning of line or indented
//    2 - Beginning of line only

INPUT="TeX"
  MODE="TeX"
  MAX_TREE_DEPTH=9
  OPEN_ON_DBL_CLICK=1
  BEGINNING_OF_LINE_ONLY=1
  CASE_SENSITIVE=1
  FILE_MUST_EXIST=1
  ALLOW_RECURSIVE_INCLUSION=0
  ALLOW_MUTIPLE_SIBLINGS_INCLUSION=0
  ALLOW_MULTIPLE_INCLUSION=1
  INCLUDE_TYPE_FILTER="tex;ltx;texi;dtx;ins;bib;bbl;txt"
  RECURSE_TYPE_FILTER="tex;ltx;texi;dtx;ins;bib;bbl;txt"
  DIRECTIVE="\include{""?""}"
  DIRECTIVE="\input{""?""}"
  DIRECTIVE="\input ""?"""
  DIRECTIVE="\include{?}"
  DIRECTIVE="\input{?}"
  DIRECTIVE="\input ?"
  // Extended Input for bibliography (expands any TeX \PATH specifications!)
  // Use PATH sparingly as it may take a long time to build the tree otherwise...
  // {??}: allows a list in up to two lines - extended it if required {????}
  DIRECTIVE="\bibliography{??}"
    FILENAME="%[!|Exe('%b\Macros\GetBibFiles.edt');|];%$('_ExTreeInput');"
    SEPARATOR=","       // allows comma separated list of bib files
    EXTENSION=".;.bib"  // possible extensions (.=default [.bib] as appended by the above macro)
    PATH="%P;%p"        // add additional locations for your bib files, eg.:
                        //     "%P;%p;%$('TeX-Local');\bibtex;C:\TeX\bibtex\My Bib Files"
  // Extended Input for import.sty (example)
  DIRECTIVE="\includefrom{?}{?}"
    FILENAME="%[!|Exe('%b\Macros\IncludeFrom.edt');|];%$('_ExTreeInput');"
  // Gather Input Directives (Comments)
  DIRECTIVE="%input ""?"""
  // Obsolete: For (<5.4) compatibility's sake ...
  DIRECTIVE="%GATHER{?}"
  DIRECTIVE="\INPUT{?}"
  DIRECTIVE="\Input{?}"

    etc ...
```

This settings can be modified (with some care). After changes you must execute `Macros -> Run Startup Macro` or `Options -> Advanced -> Load` menu command to make the changes effective.

If you type `\cite{}` you activate an *"Active String"* which displays a list of sorted citations in a special interface. Double-clicking any of the items displays the bib record while pressing Enter inserts the citation between curly brackets: try it!



Bibliography Items (40)

Dav88

AAB95
Aup91
Con78
Con90
Dau75
Dav88
dB59
dB93

Enter inserts the selected citation, while ESC closes the interface without taking any further action.

GDI Popup interface also responds to the following shortcuts:

```
F2 - Toggle the Visibility of Extra Buttons
F3 - Track Source File
F4 - Return to the Insertion Point
F5 - Refresh Data
F6 - Toggle Sorted Property (Sort/Unsort)
Enter - Return and Insert the Selected Item
ESC   - Return and exit without inserting the Selected Item
```

IMPORTANT:

The most common reason for the active string `\cite{}` to "stop working" is that WinEdt does not know where to collect bib items. For example the Main File might not have been set of else bibliography file is located in a folder that WinEdt is not aware of.

WinEdt builds the tree based on Input Directives (as specified in the `Options -> Advanced -> Tree` interface: `Tree.ini` file). Directives are mode-sensitive (since different rules apply to different document formats).

Bibliography files can be placed in the same folder as the Main File. However, since such data is usually shared between your projects you might do better to create a global bibliography and place it in your BibTeX folder in the local texmf tree. TeX's Manual explains how to do this and describes any other actions that should be taken. Eg., MiKTeX's Users have to use MiKTeX's Options Interface to refresh its FNDB (the filename database). Note that WinEdt is not aware of the folders where your TeX accessories look for input or bib files. If you decide to use the global Bibliography Data base you should specify this folder as a part of a search path for Input Directive `\bibliography{?}`. This is done by modifying the Input Directive Section in Tree.ini (`Options -> Advanced: Tree`). For example:

```
DIRECTIVE="\bibliography{??}"
  FILENAME="%[!|Exe('%b\Macros\GetBibFiles.edt');|];%$('_ExTreeInput');"
  // allows comma separated list of bib files
  SEPARATOR=","
  // possible extensions (.=default [.bib] as appended by the above macro)
  EXTENSION=".;.bib"
  // add additional locations for your bib files, eg.:
  // "%P;%p;%$('TeX-Local');\bibtex;C:\TeX\bibtex\My Bib Files"
  PATH="%P;%p;C:\localtexmf\bibtex\myBib.bib"
```

The new extended Input Directives in WinEdt 5.5 make the old "dummy" input directives redundant (as long as Tree.ini is properly configured). However, it is still possible to use a dummy input directive:

```
%Included for Gather Purpose only:
%input "C:\localtexmf\bibtex\myBib.bib"
```

Since TeX treats this line as a comment, this does not reduce portability of your documents. Once again: if TeX or BibTeX cannot find any of your input files this is not a WinEdt-related problem. Consult the Manual that comes with your TeX System!

If you prefer to work on simple one-document projects without setting the main file or building the tree you may want to change the default settings in Gather.ini so as to collect bibliography items in all opened documents. WinEdt's Macro Manual describes how to further customize the Gather Interface.

If you organize your TeX project in subfolders relative to the main file you have to use UNIX-style folder specification in "proper" include directives. For example:

```
\include{chapter1/section1.tex}
```

and *not*

```
\include{chapter1\section1.tex}
```

WinEdt understands both but TeX does not "like" backslashes "\" in filenames: you must use UNIX-style forward slashes instead!

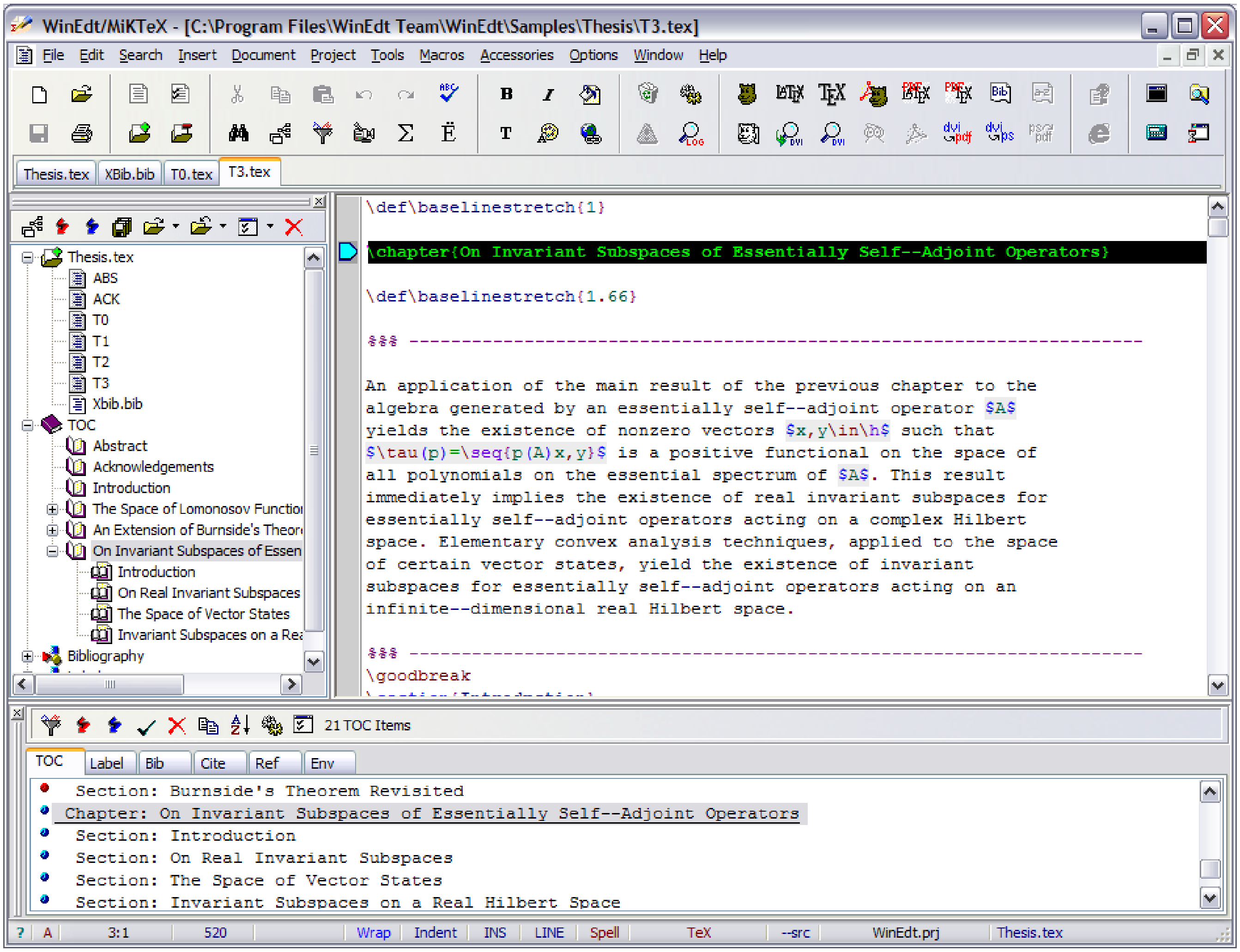
Responses such as "`\cite{}`" are defined in WinEdt as Active Strings: typing any such string triggers the execution of the associated macro. Such responses are defined in the Settings Dialog and the Help in that dialog explains the concept of *active strings* in details.

There are plenty of such strings defined but not necessary enabled in the default settings. Depending on your style of editing you may want to enable more such responses (eg. automatic insertion of closing brackets). For example if you type:

```
\begin{proof}}
```

in a TeX document the active string "`\begin{?}}`" kicks in and automatically completes your environment by inserting a properly indented "`\end{proof}`" and placing the cursor in the middle... Similarly, "`\begin{?}>`" allows you to change the name of the current environment (you are prompted for a new name and both `\begin` and `\end` are properly updated...). You don't have to be an expert in WinEdt's Macro language to enable or disable such responses or even add your custom ones based on the existing examples.

Beside the tree form and navigation bar WinEdt can also collect certain items in your project files and display them in a special "Gather" form at the bottom of your screen. Such data can be used for navigating and cross referencing.


WinEdt/MiKTeX - [C:\Program Files\WinEdt Team\WinEdt\Samples\Thesis\T3.tex]
File Edit Search Insert Document Project Tools Macros Accessories Options Window Help
Thesis.tex XBib.bib T0.tex T3.tex
Thesis.tex
ABS
ACK
T0
T1
T2
T3
Xbib.bib
TOC
Abstract
Acknowledgements
Introduction
The Space of Lomonosov Functio
An Extension of Burnside's Theor
On Invariant Subspaces of Essen
Introduction
On Real Invariant Subspaces
The Space of Vector States
Invariant Subspaces on a Rea
Bibliography
21 TOC Items
TOC Label Bib Cite Ref Env
? A 3:1 520 Wrap Indent INS LINE Spell TeX --src WinEdt.prj Thesis.tex

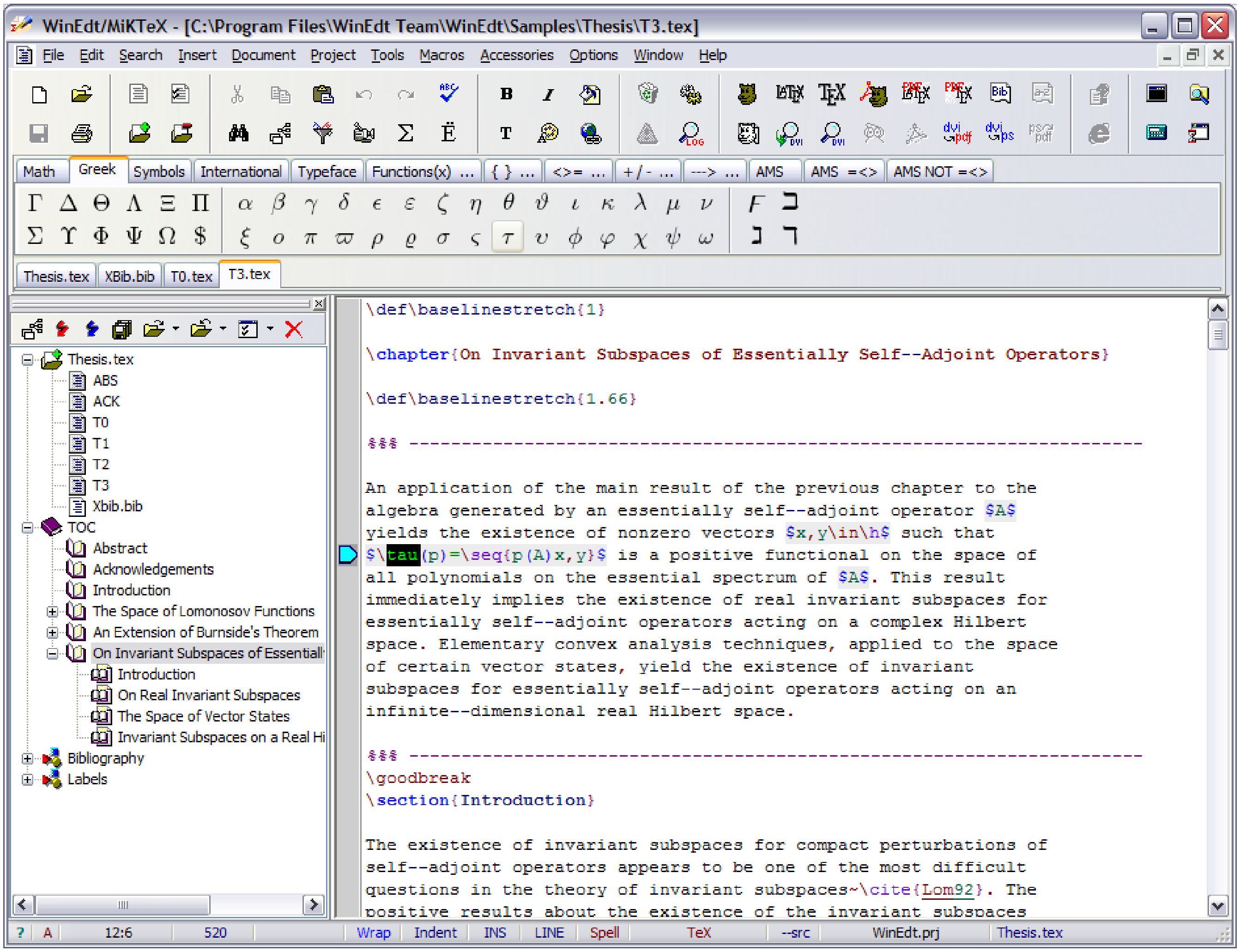
```
\def\baselinestretch{1}

\chapter{On Invariant Subspaces of Essentially Self--Adjoint Operators}

\def\baselinestretch{1.66}

%%% ---------------------------------------------------------------------

An application of the main result of the previous chapter to the
algebra generated by an essentially self--adjoint operator $A$
yields the existence of nonzero vectors $x,y\in\h$ such that
$\tau(p)=\seq{p(A)x,y}$ is a positive functional on the space of
all polynomials on the essential spectrum of $A$. This result
immediately implies the existence of real invariant subspaces for
essentially self--adjoint operators acting on a complex Hilbert
space. Elementary convex analysis techniques, applied to the space
of certain vector states, yield the existence of invariant
subspaces for essentially self--adjoint operators acting on an
infinite--dimensional real Hilbert space.

%%% ---------------------------------------------------------------------
\goodbreak
```

```
● Section: Burnside's Theorem Revisited
● Chapter: On Invariant Subspaces of Essentially Self--Adjoint Operators
●  Section: Introduction
●  Section: On Real Invariant Subspaces
●  Section: The Space of Vector States
●  Section: Invariant Subspaces on a Real Hilbert Space
```

Gather Interface is organized according to the Gather directives, which are specified in the (advanced) configuration file `Gather.ini` (`Options -> Advanced: Gather`). Gather Directives are mode-dependent to allow for different rules to be applied for different types of documents.

WinEdt's Macro Manual describes the particulars of customizing the Gather Interface....

If you press the "`Sigma`" button in the default Toolbar WinEdt displays a special GUI consisting of over 500 LaTeX symbols (organized in a few pages). Pressing the button again hides the control:

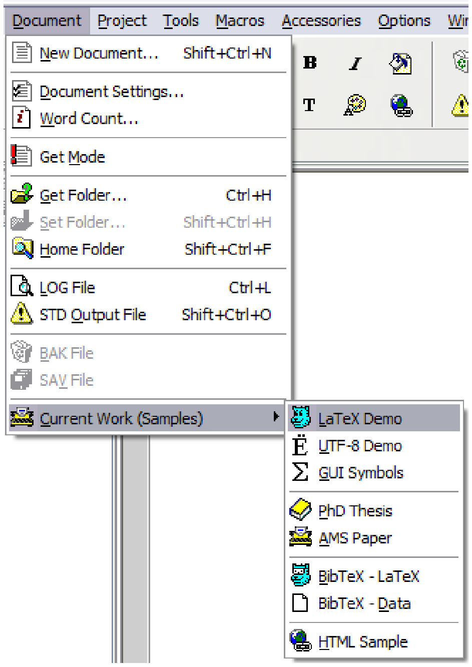

WinEdt/MiKTeX - [C:\Program Files\WinEdt Team\WinEdt\Samples\Thesis\T3.tex]
File Edit Search Insert Document Project Tools Macros Accessories Options Window Help
Math Greek Symbols International Typeface Functions(x) ... { } ... <>= ... +/- ... ---> ... AMS AMS =<> AMS NOT =<>
Γ Δ Θ Λ Ξ Π α β γ δ ϵ ε ζ η θ ϑ ι κ λ μ ν Ϝ ℶ
Σ Υ Φ Ψ Ω $ ξ o π ϖ ρ ϱ σ ς τ υ ϕ φ χ ψ ω ℷ ℸ
Thesis.tex XBib.bib T0.tex T3.tex
Thesis.tex
ABS
ACK
T0
T1
T2
T3
Xbib.bib
TOC
Abstract
Acknowledgements
Introduction
The Space of Lomonosov Functions
An Extension of Burnside's Theorem
On Invariant Subspaces of Essentiall
Introduction
On Real Invariant Subspaces
The Space of Vector States
Invariant Subspaces on a Real Hi
Bibliography
Labels
? A 12:6 520 Wrap Indent INS LINE Spell TeX --src WinEdt.prj Thesis.tex

```
\def\baselinestretch{1}

\chapter{On Invariant Subspaces of Essentially Self--Adjoint Operators}

\def\baselinestretch{1.66}

%%% ----------------------------------------------------------------------

An application of the main result of the previous chapter to the
algebra generated by an essentially self--adjoint operator $A$
yields the existence of nonzero vectors $x,y\in\h$ such that
$\tau(p)=\seq{p(A)x,y}$ is a positive functional on the space of
all polynomials on the essential spectrum of $A$. This result
immediately implies the existence of real invariant subspaces for
essentially self--adjoint operators acting on a complex Hilbert
space. Elementary convex analysis techniques, applied to the space
of certain vector states, yield the existence of invariant
subspaces for essentially self--adjoint operators acting on an
infinite--dimensional real Hilbert space.

%%% ----------------------------------------------------------------------
\goodbreak
\section{Introduction}

The existence of invariant subspaces for compact perturbations of
self--adjoint operators appears to be one of the most difficult
questions in the theory of invariant subspaces~\cite{Lom92}. The
positive results about the existence of the invariant subspaces
```

Important: if you want to use all the symbols in your LaTeX Document you should load the `amsmath` package in your preamble:

```
\usepackage{amsmath}
\usepackage{amsxtra}
\usepackage{amstext}
\usepackage{amssymb}
\usepackage{latexsym}
\usepackage{dsfont} % for \mathds{N}
```

The file "`_Gui Source.tex`" in WinEdt's `Bitmaps\GUI` folder describes how the images were generated (it is time-consuming but it can be customized)... The file can be opened through `Documents -> Current Work (Samples) -> GUI Symbols` menu command.



Graphics inclusion in LaTeX documents is *not* a WinEdt-related issue. You are wasting your (and our) time if you are writing with such questions. However, the sample `LaTeX Demo` in the above menu illustrates and describes how it can be done. Assuming that you have a properly installed MiKTeX and Ghostscript, the sample compiles into dvi, ps, or pdf by any available procedure in the default settings (try it!). `LaTeX Demo` also contains useful hints when it comes to TeX and international characters and other [La]TeX related issues. If you need more help you'll have to consult the resources and documentation on www.tug.org where you'll also find recommended literature that will make your TeX-ing enjoyable and stress-free...