

## Настройка используемого программного обеспечения

### 1. Программное обеспечение для Лиспа

Используемое свободно распространяемое программное обеспечение состоит из интерпретатора и интегрированной оболочки.

Интерпретатор GNU Clisp 2.49 можно скачать по ссылке <http://sourceforge.net/projects/clisp/files/clisp/>.

Это самораспаковывающийся архив 4,4 Мб.

GNU Clisp реализован немецкими студентами Бруно Хайбле (Bruno Haible) и Михаэлем Штоллем (Michael Stoll) в 1992 г. Он соответствует ANSI Common Lisp стандарту, работает под Unix, Windows. Для запуска интерпретатора следует запустить файл clisp.exe.

Консольный интерфейс GNU Clisp приведен на рис.1.

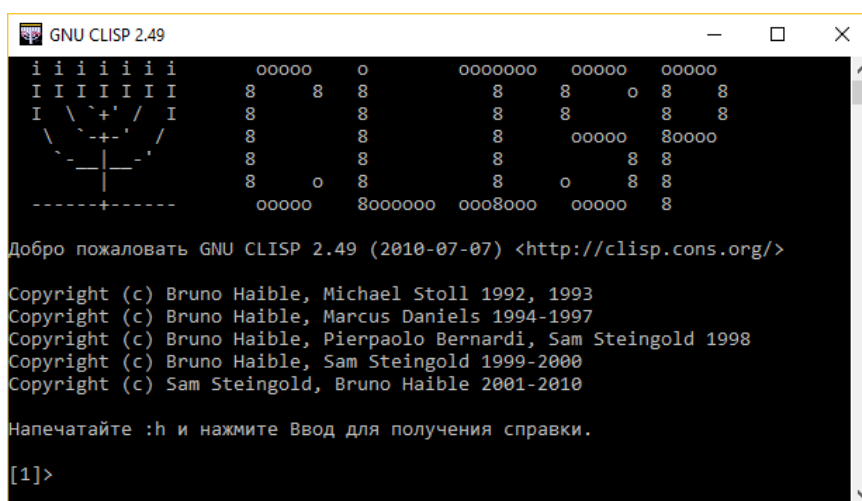


Рис. 2. Интерфейс интерпретатора GNU Clisp

Значок «>» является приглашением для ввода команды. Когда пользователь заканчивает ввод вызова функции или выражения, то интерпретатор сразу же вычисляет значение этого выражения, выдает на экран это значение и очередное приглашение для ввода команды. Если при выполнении команды появилась ошибка, то предлагаются варианты продолжения. В том числе, вернуться на предыдущий уровень (до ошибки) можно с помощью команды abort. Для завершения работы следует ввести команду (exit) или (bye).

GNU Clisp представляет собой полноценную среду для написания, отладки и исполнения программ на языке Lisp, но как и любое консольное приложение, возможности встроенного редактора сильно ограничены, именно поэтому мы будем использовать оболочку LispIDE для GNU Clisp, которая предоставляет дополнительные удобные возможности, в первую очередь для редактирования.

Редактор LispIDE можно скачать по ссылке <http://daansystems.com/lispide/> – установщик (архив для скачивания находится внизу страницы). После запуска редактора первый раз необходимо в открывшемся окне прописать путь к файлу clisp.exe (рис.2).

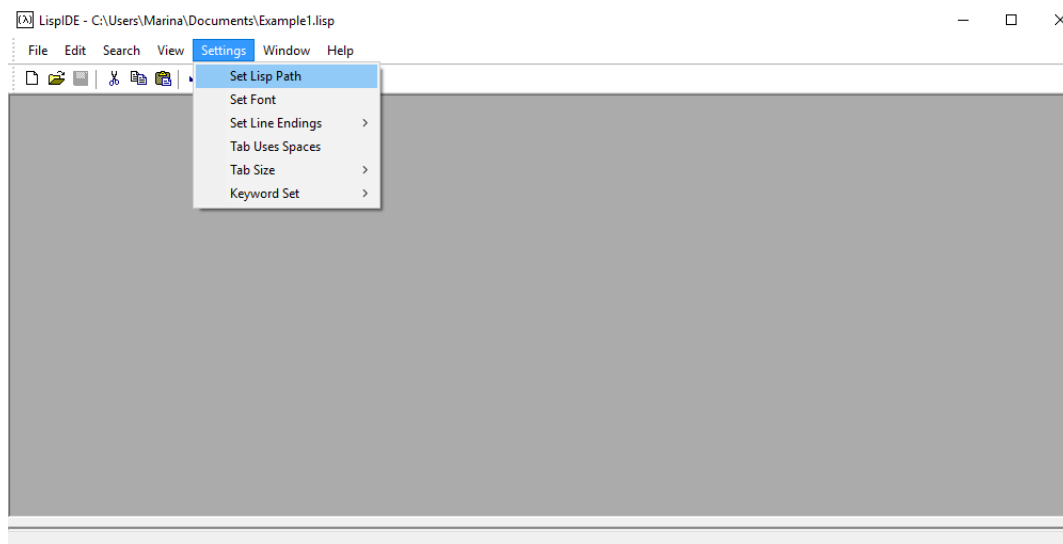


Рис. 2. Установка пути к интерпретатору Clisp (шаг 1)

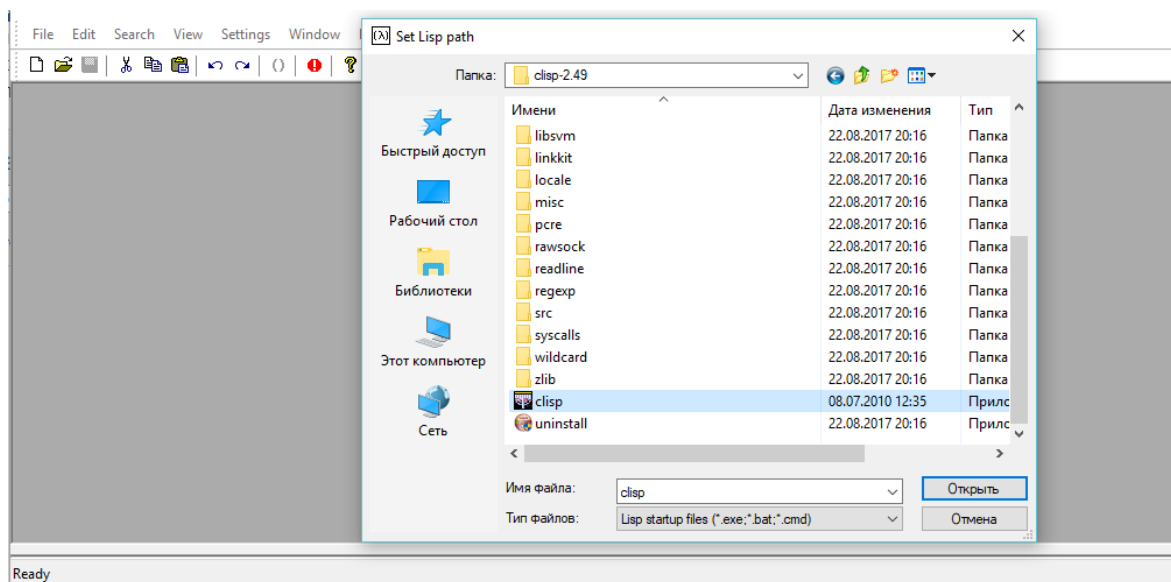
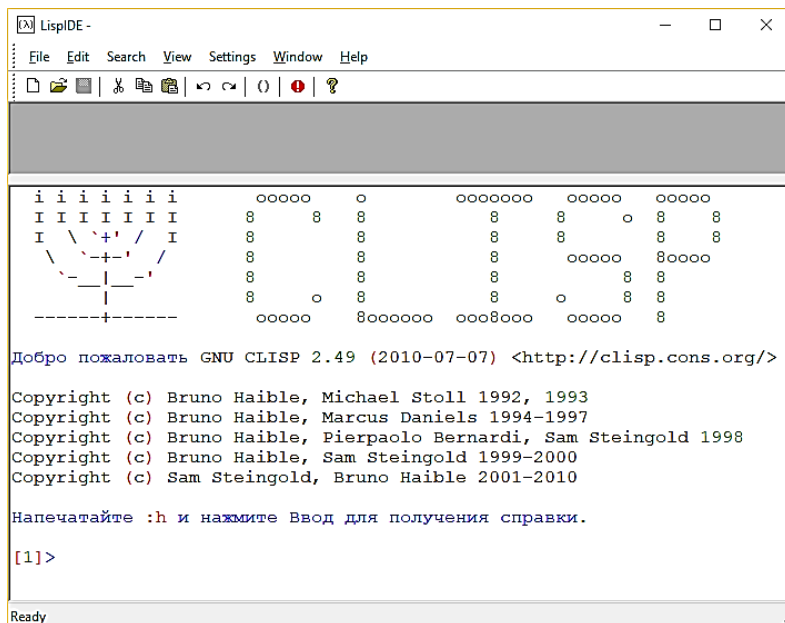


Рис. 3. Установка пути к интерпретатору Clisp (шаг 2)

В результате окно LispIDE должно принять вид, изображенный на рис. 4.



### Внимание!

Если нижнее окно (окно интерпретатора Clisp) не отображается, то необходимо при зажатой левой клавише мыши над горизонтальным разделителем рабочих областей перетащить его вверх.

Рис. 4. Окно LispIDE с окном интерпретатора Clisp

Далее открываем файл с программой на Clisp или создаем новый, выбрав соответствующую пиктограмму на панели инструментов. На рис. 5 изображено окно создания нового файла.

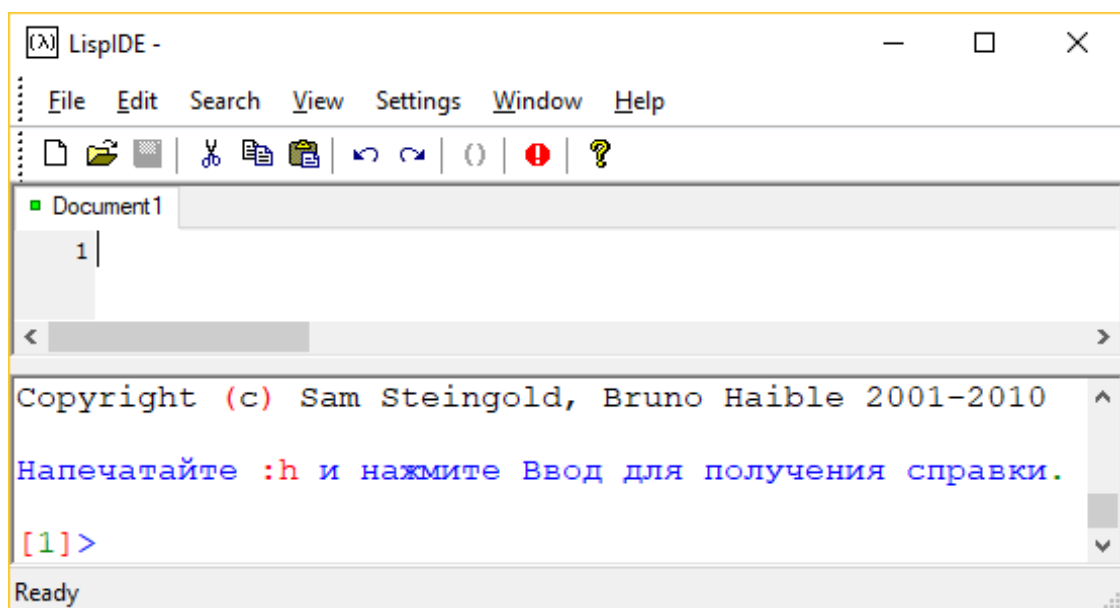


Рис. 5. Окно LispIDE с окнами редактора нового файла и интерпретатора Clisp

*Замечание:* LispIDE не позволяет сохранять в папки или файлы с названиями на русском языке.

## 2. Программное обеспечение для Пролога

Также, как и в случае с Лиспом, используемое свободно распространяемое программное обеспечение состоит из интерпретатора и интегрированной оболочки.

Интерпретатор SWI-Prolog 7.6.4 (или новее) можно скачать по ссылке (скачиваем 32-битную версию вне зависимости от разрядности системы):

<http://www.swi-prolog.org/download/stable> – установщик.

Консольный интерфейс SWI-Prolog изображен на рис.6.

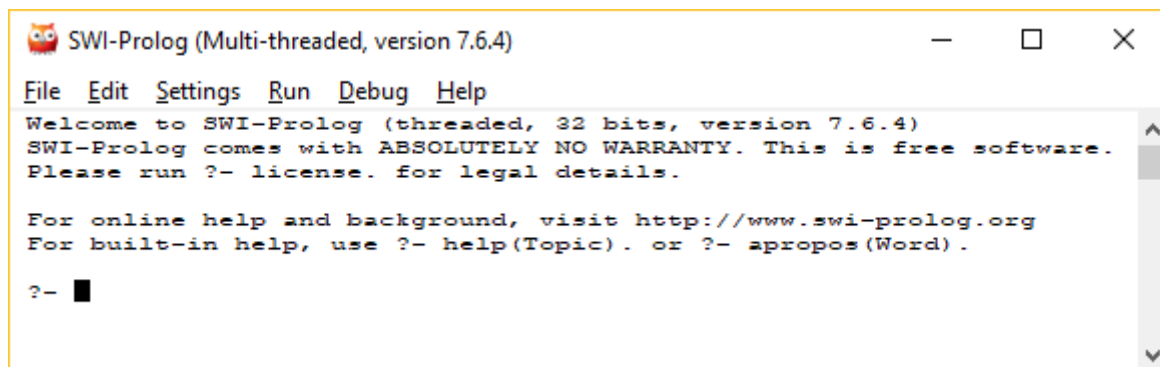


Рис. 6. Окно интерпретатора SWI-Prolog

SWI-Prolog представляет собой полноценную среду для написания, отладки и исполнения программ на языке SWI-Prolog. Встроенный редактор открывается в новом окне при выборе в меню File пункта редактирования файла или создания нового (рис.7).

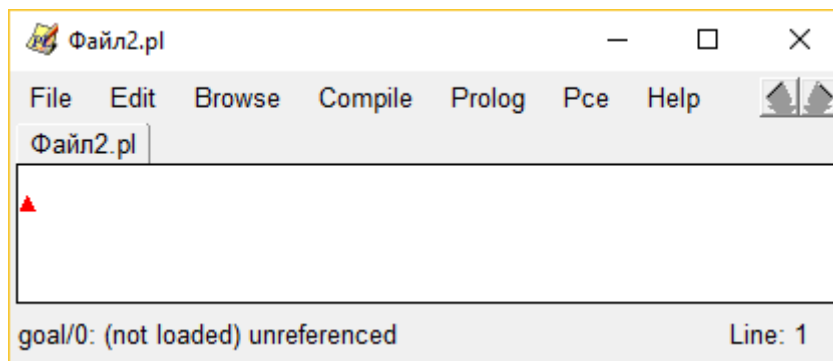


Рис. 7. Окно редактора SWI-Prolog

Запуск программы на выполнение из окна редактора осуществляется выбором Compile-Compile buffer. Далее в окне интерпретатора вводится вызов предиката, и далее работа идет с двумя окнами.

Далее вместо встроенного редактора будет использоваться оболочка SWI-Prolog-Editor 4.26, которая представляет дополнительные удобные возможности. Редактор SWI-Prolog-Editor 4.26 (Может работать только с 32-битной версией SWI-Prolog) можно скачать по ссылке со страницы <http://www.frankengel.org/index.php/schule/informatik/informatik-software/145-informatik-software>.

После запуска установщика SWI-Prolog-Editor на первом шаге выберите сохранение настроек в INI-файлах и, нажав кнопку Язык, в появившемся окне выберите russian (рис. 8 – 9).

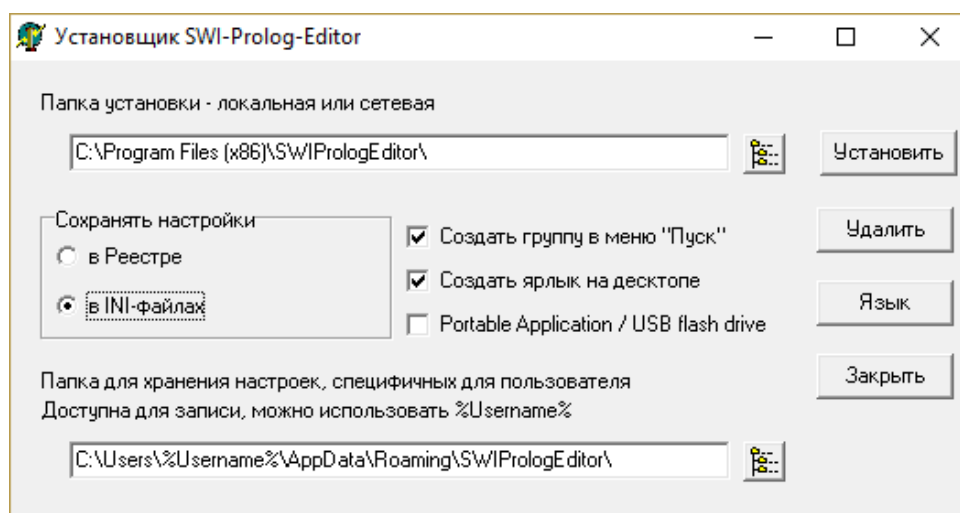


Рис. 8. Окно установки SWI-Prolog-Editor

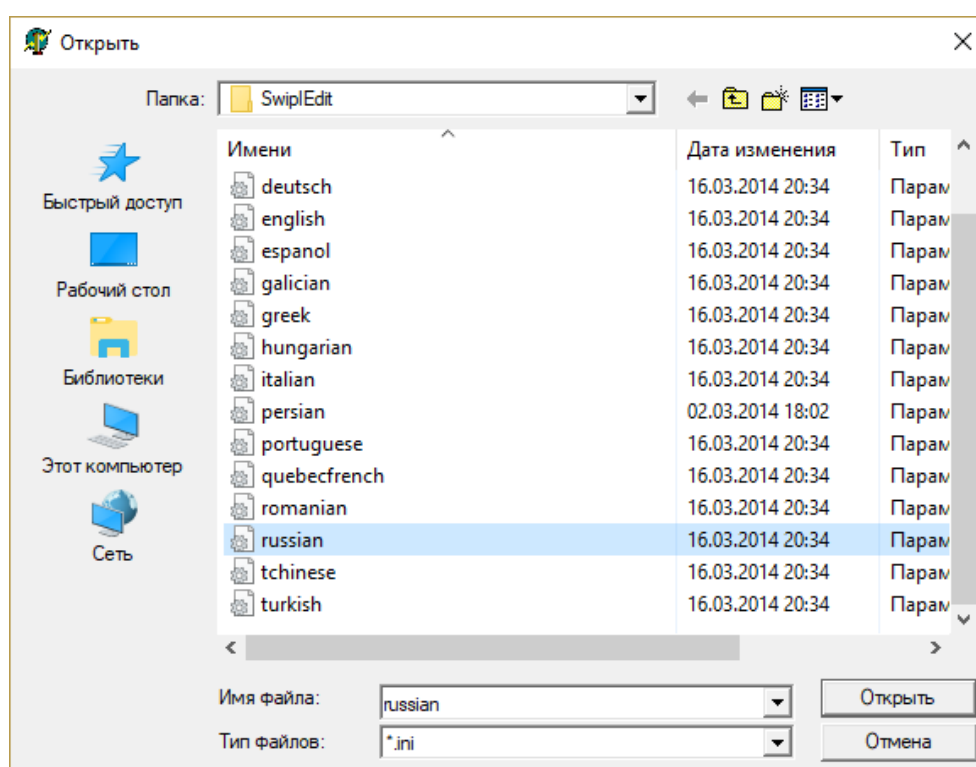


Рис. 9. Выбор языка при установке SWI-Prolog-Editor

На рис. 10 изображена интегрированная среда SWI-Prolog-Editor, запущенная после установки.

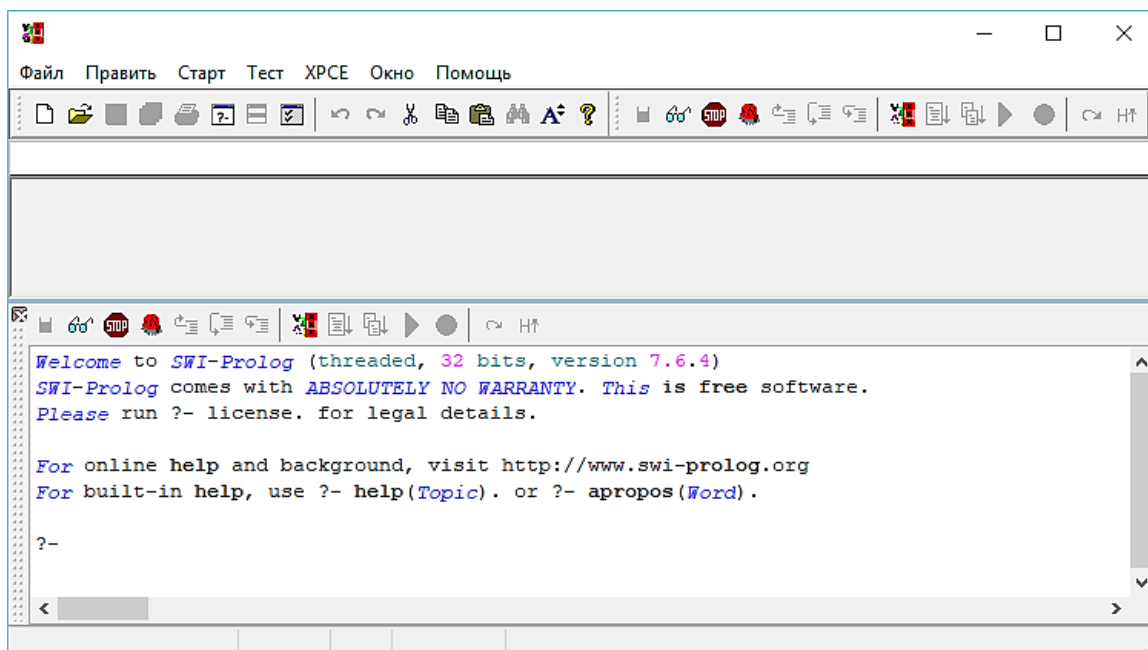


Рис. 10. Интегрированная среда SWI-Prolog-Editor с окном SWI-Prolog

Если связывание SWI-Prolog-Editor с SWI-Prolog не произошло автоматически, то необходимо указать путь к исполняемому файлу SWI-Prolog. Для этого выберите пункт главного меню Окно – Конфигурация и в открывшемся окне пропишите путь к папке bin, в которой находится исполняемый файл swipl.exe (рис. 11 – 12).

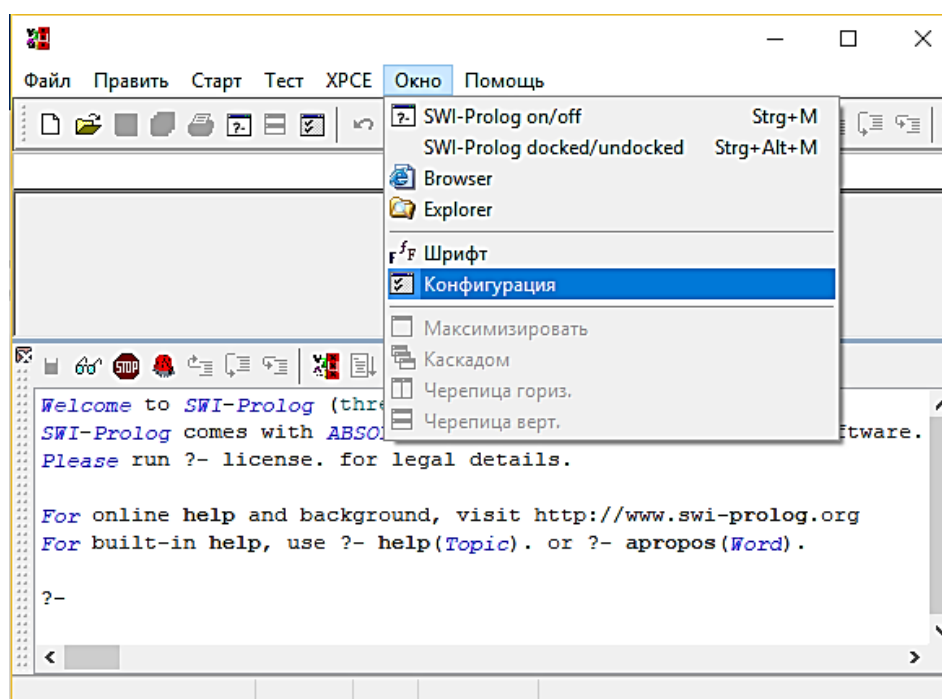


Рис. 11. Настройка связи SWI-Prolog-Editor с интерпретатором SWI-Prolog (шаг 1)

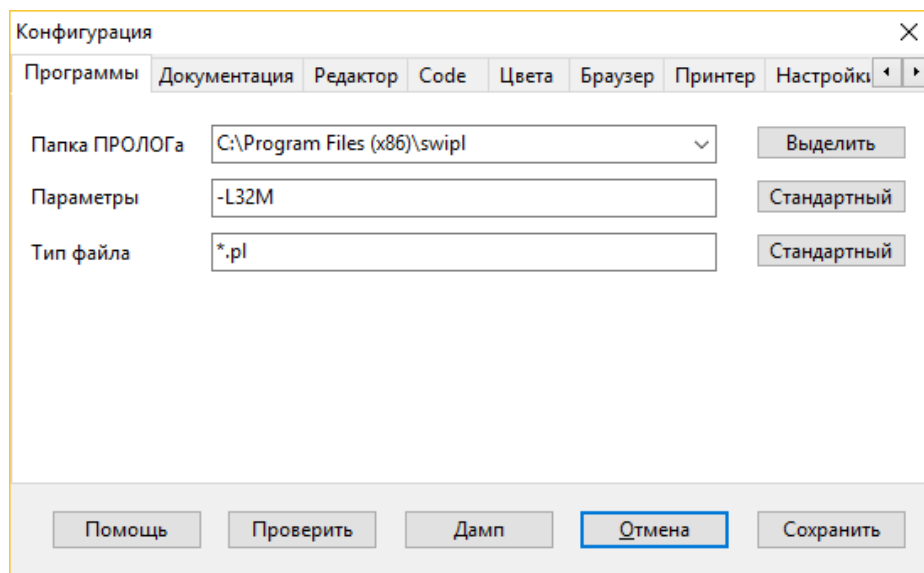


Рис. 12. Настройка связи SWI-Prolog-Editor с интерпретатором SWI-Prolog (шаг 2)

Настройка кодовой страницы необходима для правильного сопоставления строковых констант, набранных русским алфавитом, между текстом программы в среде SWI-Prolog-Editor и языком SWI-Prolog. Для настройки в этом же окне установите номер кодовой страницы 1251 и сохраните выбранные настройки, нажав кнопку Сохранить (рис. 13).

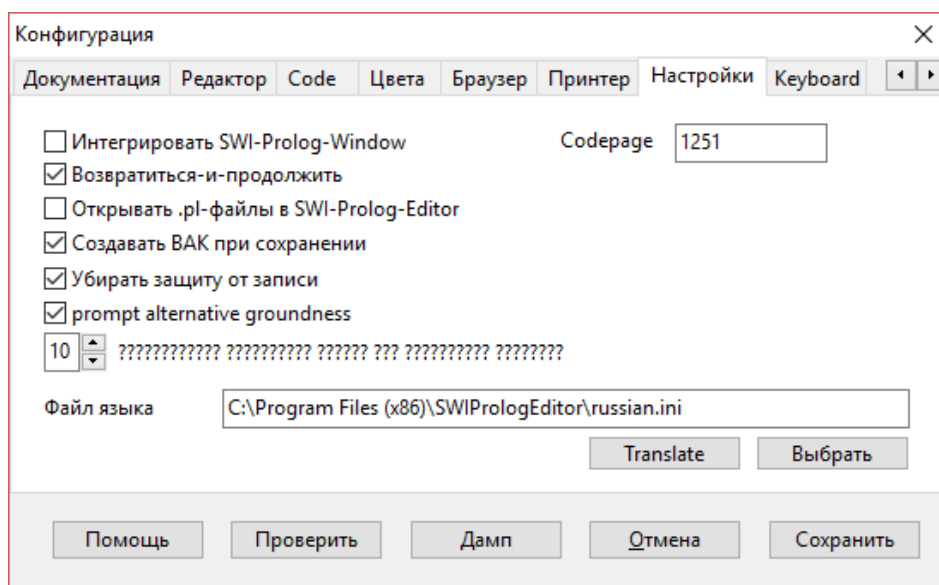


Рис. 13. Настройка кодовой страницы

Далее открываем файл с программой на SWI-Prolog или создаем новый, выбрав соответствующую пиктограмму на панели инструментов (рис.14).

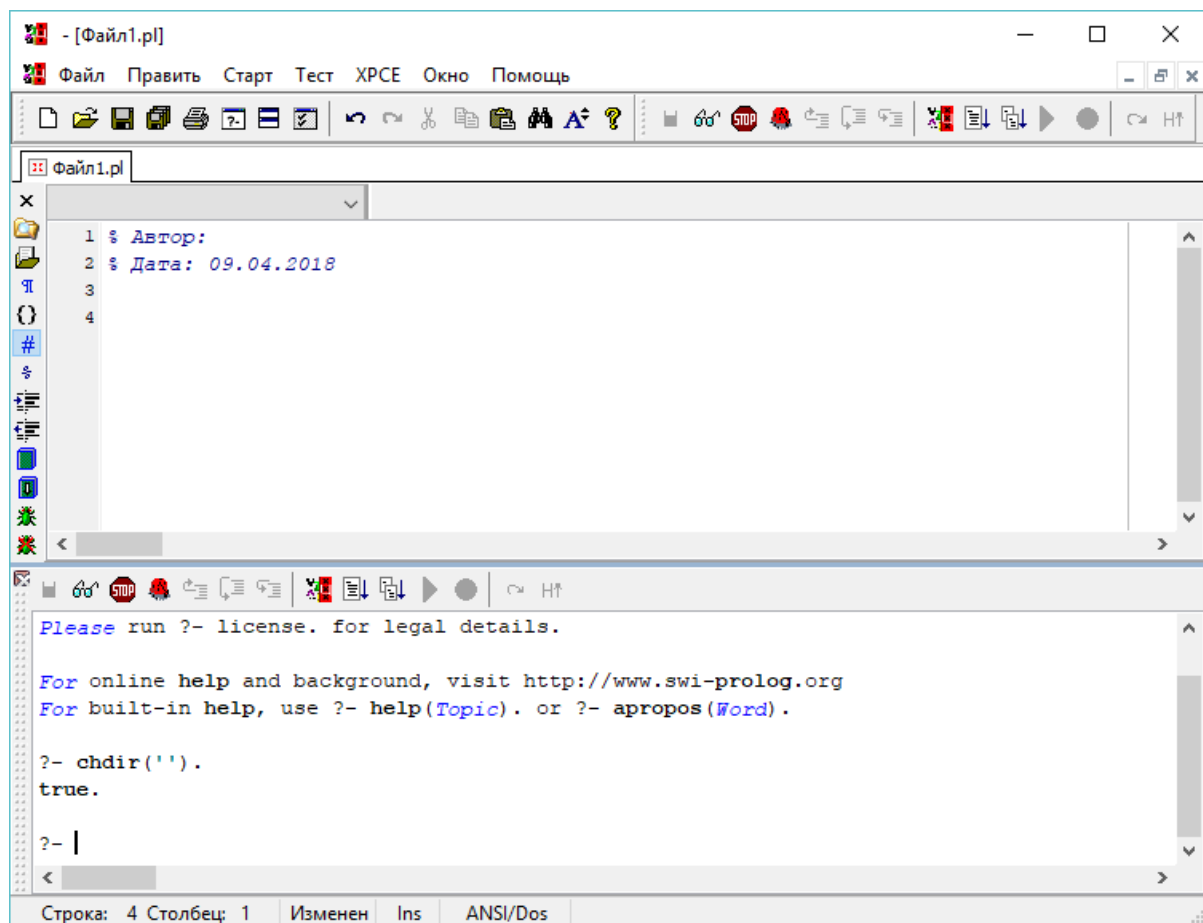


Рис. 14. Интегрированная среда SWI-Prolog-Editor с окном SWI-Prolog и окном редактирования нового файла