

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САМАРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ИМЕНИ АКАДЕМИКА С.П. КОРОЛЕВА»

Институт информатики, математики и электроники
Факультет информатики
Кафедра технической кибернетики

Параллельное программирование

Методические указания
к лабораторной работе №1

САМАРА 2017

Составитель: доц., Козлова Е.С.

УДК 519.681

Методические указания к лабораторным работам

Самарский государственный аэрокосмический университет
имени академика С.П.КОРОЛЁВА (национальный исследовательский университет)

Составитель: Е.С. Козлова

Самара, 2018. 12 с.

Методические указания предназначены для бакалавров направления 010400.62
«Прикладная математика и информатика»

Печатается по решению редакционно-издательского совета Самарского
университета

Рецензент:

1 Работа на кластере

1.1 Удаленный доступ к командной строке сервера

Доступ к серверам осуществляется через протокол SSH (собственно SSH – «secure shell»). Для работы с командной строкой рекомендуется программа **PuTTY**. Эта программа позволяет единожды ввести адрес сервера и, в дальнейшем, выбирать его из списка сессий. В настройках **PuTTY** необходимо указать протокол SSH. При первом соединении с сервером программа выдает предупреждение о том, что ключ шифрования сервера ранее не использовался, и предлагает его сохранить или отвергнуть (продолжить сеанс или прервать). Необходимо выбрать продолжение сеанса.

Для установки программы **PuTTY** на свой компьютер воспользуйтесь ресурсами <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>.

Чтобы войти на нужный кластер, необходимо:

1. Запустить программу **PuTTY**.

2. В открытом окне ввести имя (Host Name) или IP-адрес кластера. В нашем случае вводим: **sk.ssau.ru**.

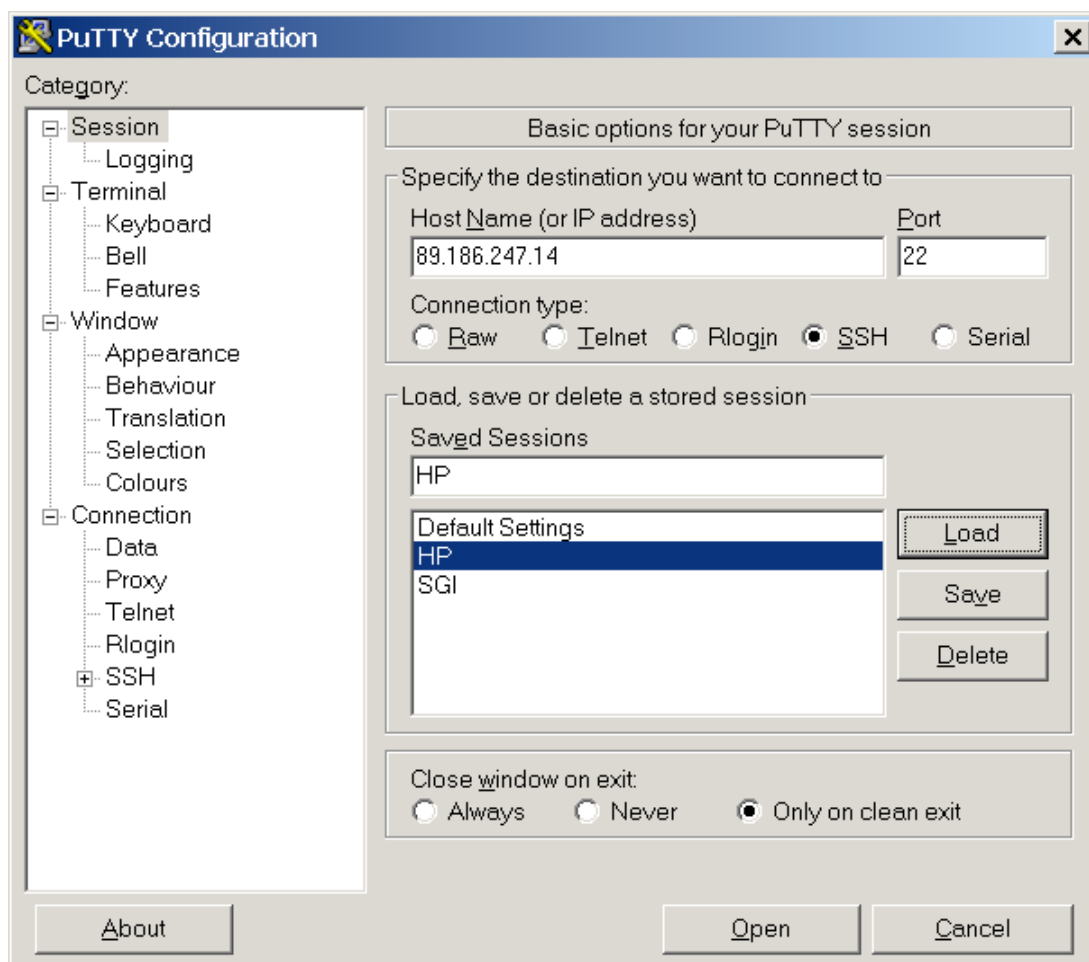


Рис. 1.1 Диалоговое окно PuTTY

3. Выполнить команду **OPEN**.

Откроется окно , в котором после ввода логина и пароля пользователь может работать на кластере из командной строки или используя **Midnight Commander**.

Для удобства работы можно выполнить некоторые настройки и сохранить их для дальнейших сеансов.

Например, чтобы при входе на кластер пользователь мог прочитать вводный текст на русском языке, необходимо выбрать кодировку UTF – 8 и сохранить данную настройку, проделав следующие шаги:

1. Открыть окно **PuTTY**:
2. Выбрать раздел **Translation**.

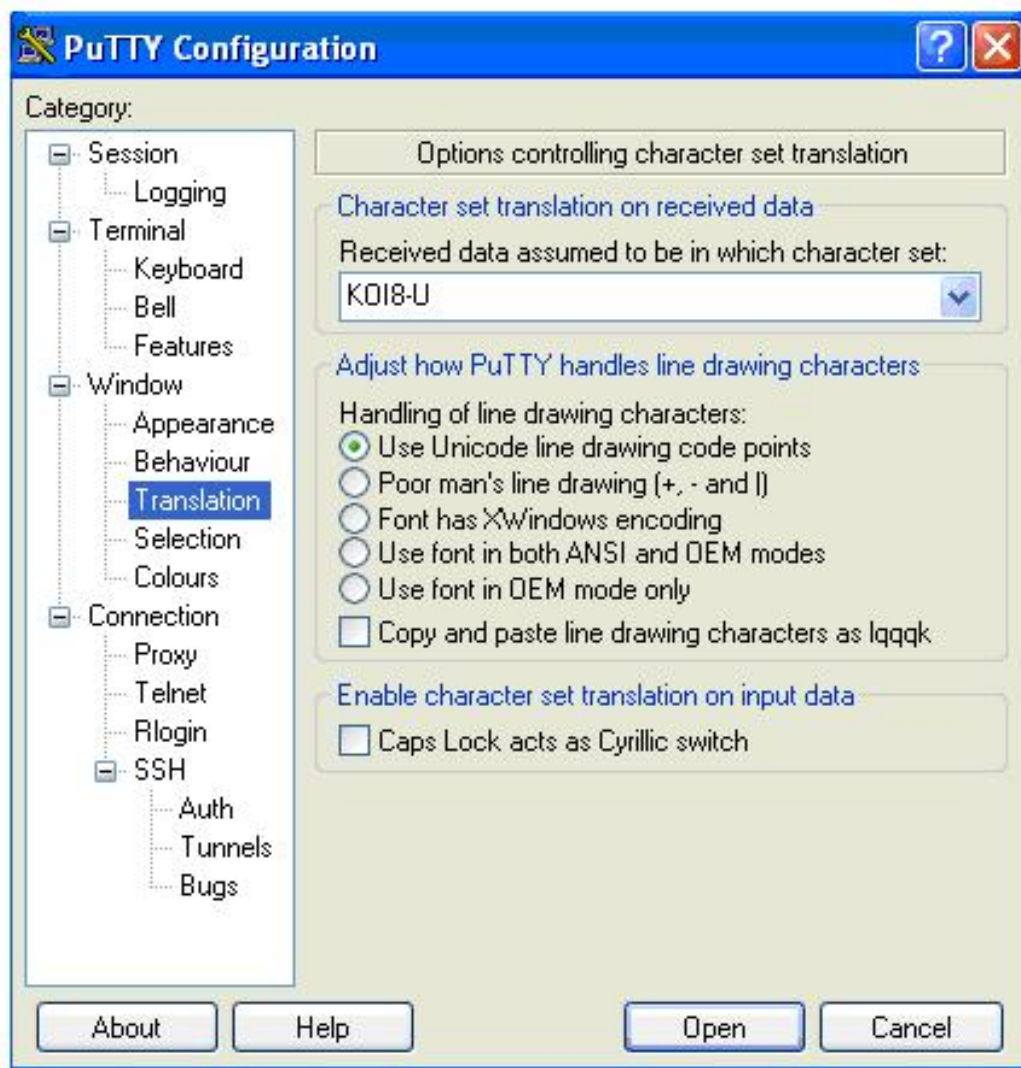


Рис. 1.2 Выбор окна трансляции

3. Выбрать кодировку UTF – 8.
4. Перейти к разделу session.
5. Ввести имя кластера в окна Host Name и Saved Sessions.
6. Нажать клавишу Save (сохранить).
7. Нажать клавишу Open .

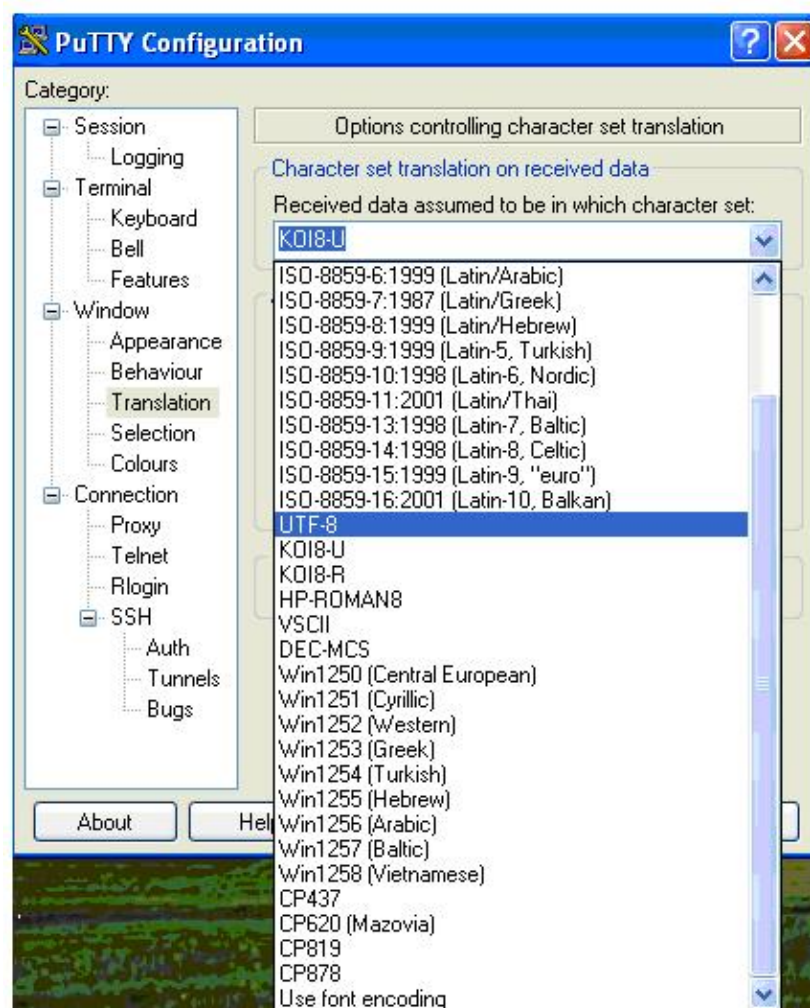


Рис. 1.3 Выбор кодировки

1.2 Обмен файлами между компьютером и сервером

Для обмена файлами между компьютером и сервером рекомендуется использовать программу WinSCP.

При копировании файлов (протоколы SCP – «secure copy» и SFTP – «secure ftp») в настройках WinSCP рекомендуется выбрать протокол SFTP. Программа WinSCP позволяет сохранить имя пользователя и пароль. При первом соединении с сервером программа выдает предупреждение о том, что ключ шифрования сервера ранее не использовался, и предлагает его сохранить или отвергнуть (продолжить сеанс или прервать). Надо выбрать продолжение сеанса. Программа WinSCP может использоваться как самостоятельное приложение и как плагин для других программ проводников, например для Far Commander.

Тексты программ и данные для задачи можно записать в отдельный каталог на сервере, тогда вся информация по этой задаче будет сохраняться в этом каталоге (объектные модули после трансляции текстов программ, файл с сообщениями об ошибках, выходные файлы).

На сервере существует понятие домашнего каталога. При работе с командной строкой этот каталог становится текущим после установления терминального соединения. Программа WinSCP может запоминать последний посещенный в предыдущей сессии каталог. Поэтому важно следить, в какой каталог производится копирование.

После копирования необходимо перейти в рабочий каталог, выполнить компиляцию программы и ее запуск. Для установки программы WinSCP на свой компьютер воспользуйтесь ресурсами <http://winscp.net/eng/index.php>.

Чтобы войти на нужный кластер, необходимо:

1. Запустить программу WinSCP.

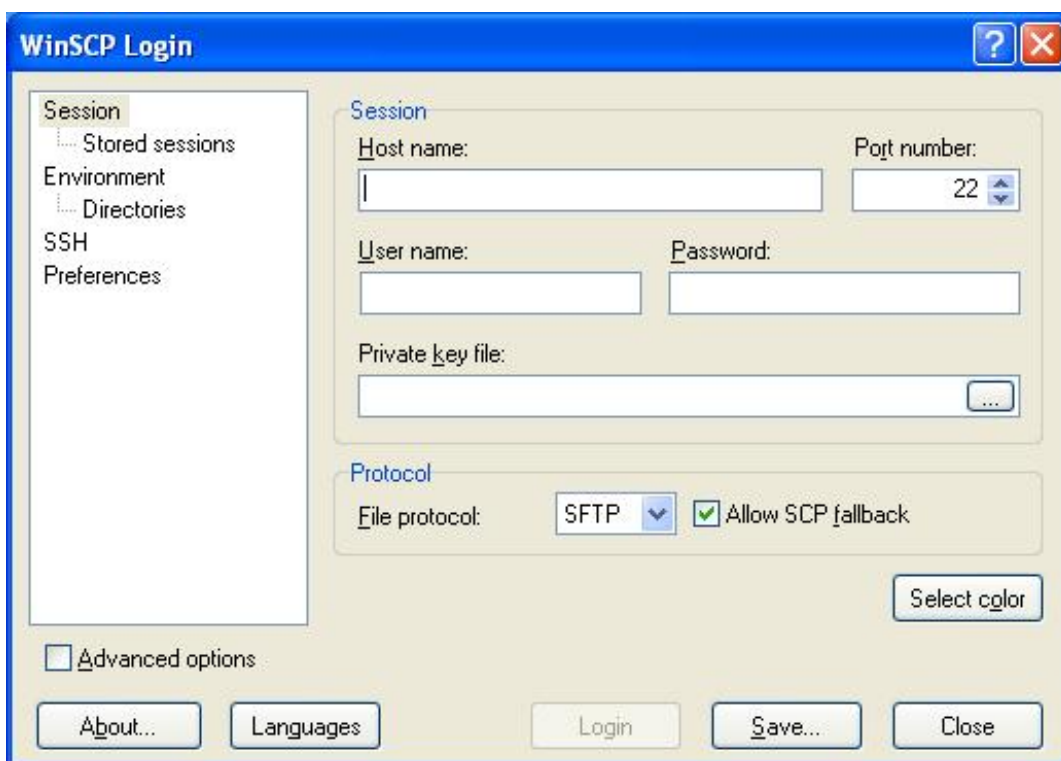


Рис. 1.4 Диалоговое окно WinSCP

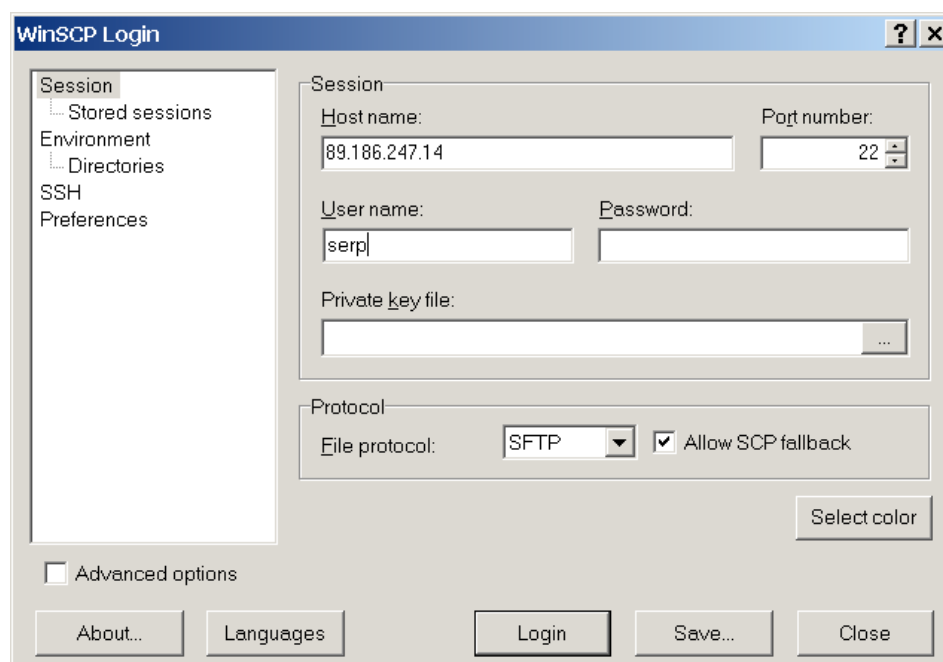


Рис. 1.5 Создание сессии в WinSCP

2. Ввести необходимую информацию для подключения к выбранному серверу.
3. Сохранить эти настройки (аналогично можно выполнить и сохранить настройки для входа на другие серверы/кластеры).

4. Выбрать сессию и выполнить **LOGIN**.

5. В новом окне подтвердить пароль.

Далее можно работать с файлами исходного компьютера и сервера, используя настроенный интерфейс.

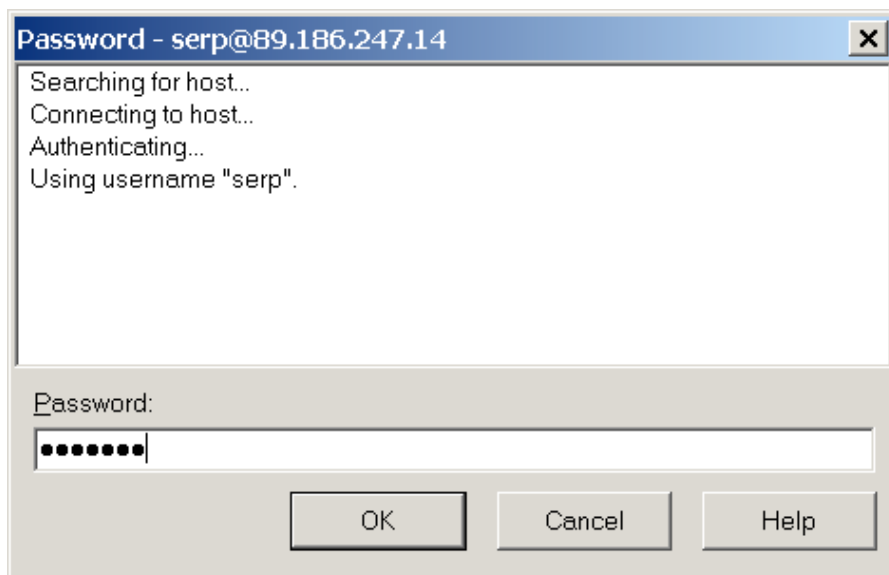


Рис. 1.6 Создание сессии в WinSCP

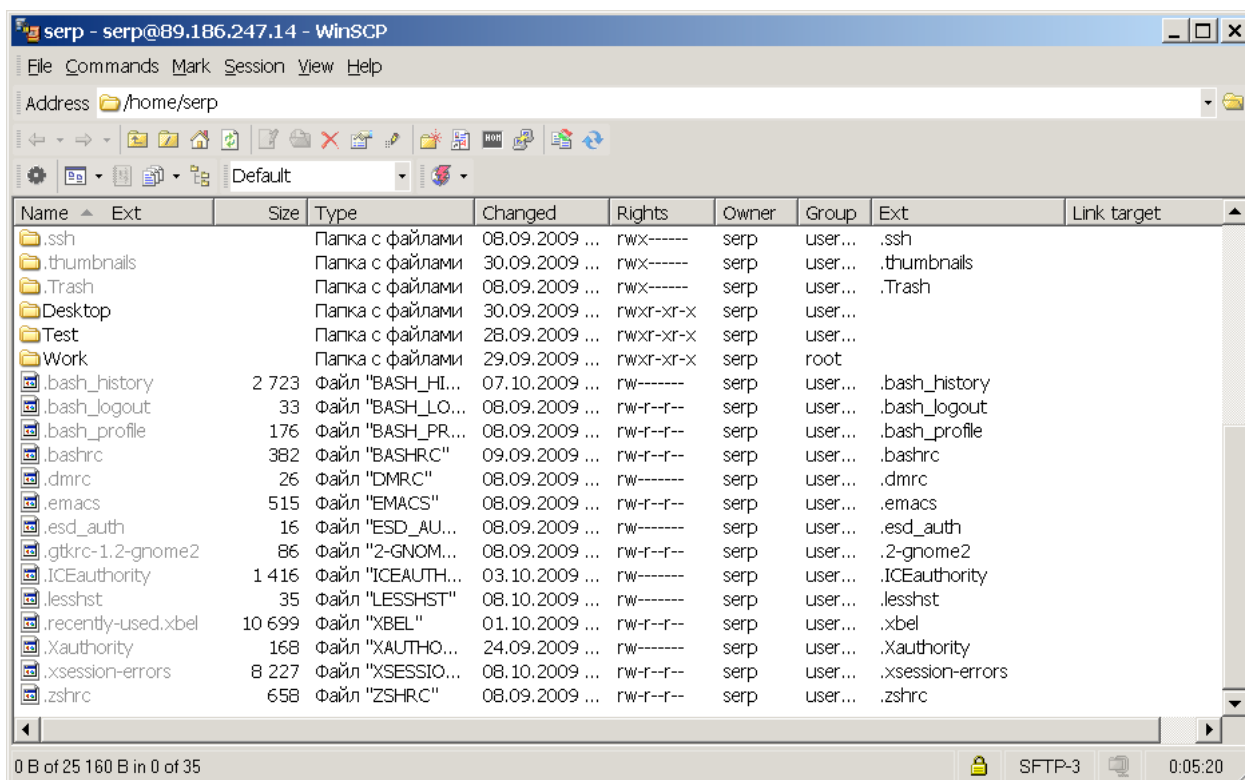


Рис. 1.7 Окно сессии WinSCP

Завершение работы с программой и прекращение связи проводится клавишей **F10** или выбором в меню пункта **Commands/Quit**. Требуется подтвердить окончание работы в окне завершения.

1.3 Редактирование текста программ пользователя

Процесс разработки программного обеспечения, независимо от размеров и предназначения программы содержит этап редактирования исходных текстов программы. Конечно, можно изменять свои программы на рабочем компьютере, а потом копировать их на кластер. Также можно редактировать файлы прямо на кластере. Для этого на кластере имеется несколько текстовых редакторов разной степени удобства: vim, ed, emacs, joe. Наиболее простым для освоения является редактор, встроенный в оболочку Midnight Commander.

Для запуска файлового менеджера Midnight Commander используется команда **mc**.

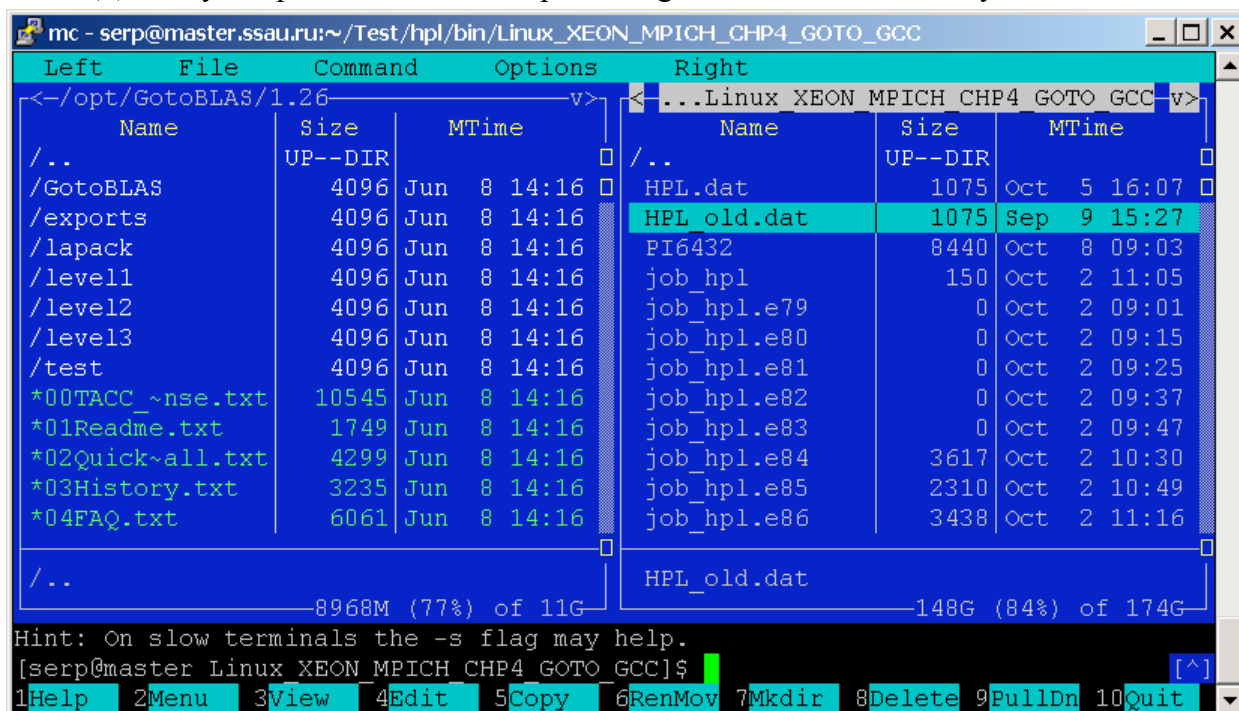


Рис. 1.8 Окно оболочки Midnight Commander

Midnight Commander — это файловый менеджер с текстовым интерфейсом. Его предназначение — упростить основные действия пользователя, связанные с управлением файлами. Принцип работы *Midnight Commander* такой же, как и у *Far Manager*, *TotalCommander*, или *Norton Commander*. Экран состоит из двух панелей, в которых отображается список файлов и каталогов в выбранных каталогах, и пользователь может выполнять некоторый набор действий над этими файлами. В нижней части экрана расположена командная строка и панель горячих клавиш **F1-F10**. Можно вызвать верхнее меню, нажав клавишу **F9**.

Одна из панелей всегда является активной, а в ней курсор установлен на активный файл. Пользователь может выполнять действия либо с активным файлом или каталогом, либо групповые операции со всеми объектами активной панели. Также доступны некоторые общие операции: поиск файлов, помощь по работе с *mc*, выполнение команд операционной системы и т.д.

Для редактирования файла необходимо клавишами управления курсором выбрать нужный файл, и нажать клавишу **F4**. Запустится редактор текстовых файлов, выйти из которого можно, нажав клавишу **F10** либо **Esc**. Чтобы сохранить изменения, необходимо нажать клавишу **F2**, либо выбрать нужный вариант при выходе из редактора.

Для создания нового файла нужно нажать **Shift+F4**. Откроется окно редактора, и при сохранении изменений, программа предложит ввести имя сохраняемого файла.

1.4 Компиляция и запуск программ пользователя

Для компиляции и сборки параллельных программ используются следующие утилиты:

- *mpicc* — для MPI-программ;
- *icc* — для OpenMP-программ.

Сначала необходимо подгрузить модули. Для этого используйте следующие команды:

```
module load intel/icc  
module load impi/4
```

Для непосредственного запуска программ используются следующие команды:

```
icc -openmp -o test omp.c  
mpicc -o test mpi.c
```

Запуск задачи на кластере выполняется с использованием системы пакетного запуска *PBS Torque*. Это гибкий и удобный способ работы на кластере, особенно в многопользовательском режиме. Перечислим только самые простые варианты использования.

Запуск задачи осуществляется следующим образом:

```
qsub <имя_скрипта>
```

Сценарий, или скрипт не обязательно должен быть исполняемым, это может быть простой текстовый файл, содержащий размещенные построчно команды в формате обычной командной строки. Кроме исполняемых команд скрипт может содержать специальные директивы **PBS Torque**, которые позволяют описать необходимые для выполнения задачи ресурсы: количество процессоров, память, время, политику многозадачного режима – и многие другие параметры с различной степенью детализации.

Torque ставит запущенную задачу в очередь, отыскивает свободные процессоры (вычислительные узлы) и запускает задачу. При этом пользователь может закрыть свою терминальную сессию – задача будет выполняться в фоновом режиме. Пользователь может просмотреть очередь при помощи команды

```
qstat
```

Команда отобразит состояние очереди задач. При этом будут отображаться задачи запущенные только этим пользователем (рис. 1.8).

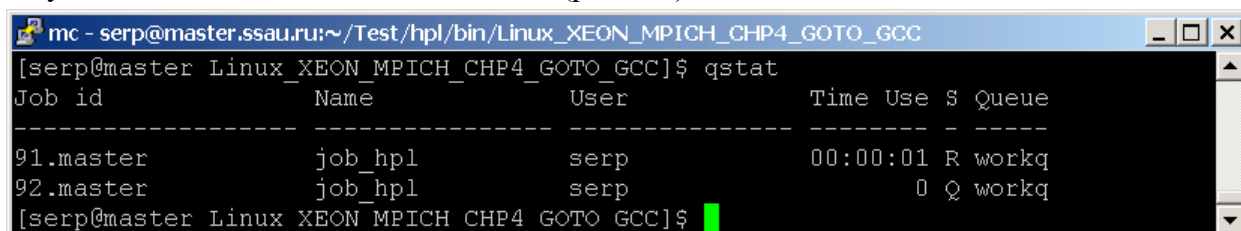


Рис. 1.8 Очередь задач пользователя

Здесь показана очередь из двух заданий с номерами 91 и 92, которые запущены пользователем *serp*, имя очереди *workq*, имя задачи *job_hpl*.

С помощью команды

qstat -nможно получить более полную информацию о количестве заказанных вычислительных узлов (колонок NDS), максимальном времени выполнения (Time), количестве задач (TSK) и состоянии (S): • Q – в очереди, • R – выполняется.

Удалить задание из очереди можно командой

```
qdel <номер_задания>.
```

2 Задание на лабораторную работу

Лабораторная работа 1

Произвести запуск программ «Hello world», которые используют технологии MPI и OpenMP, на различном количестве процессоров (потоков). В ходе анализа работы программы оценить время ее выполнения на различном количестве исполняющих нитей (процессов), для чего в заготовках необходимо произвести изменения. Ниже приведен шаблоны программ для запуска.

Пример 1. Код MPI

```
#include "mpi.h"
#include "stdio.h"
#include <time.h>
#include <math.h>
#include <omp.h>

int main(int argc, char* argv[])
{
    int rank, ranksize, i;
    MPI_Init(&argc, &argv); //
    //Определяем свой номер в группе:
    MPI_Comm_rank(MPI_COMM_WORLD, &rank); //
    //Определяем размер группы:
    MPI_Comm_size(MPI_COMM_WORLD, &ranksize); //
    printf("Hello world from process %d from total number of
%d\n", rank, ranksize);
    MPI_Finalize(); //
    return 0;
}
```

Пример 2. Код OpenMP

```
#include <math.h>
#include <omp.h>
#include <time.h>
#include <stdlib.h>
#include <locale.h>
#include <stdio.h>

int main(int argc, char* argv[])
{
    omp_set_num_threads(16);
    int nThreads, theadNum;
    #pragma omp parallel private(nThreads, theadNum)
    {
        nThreads = omp_get_num_threads();
        theadNum = omp_get_thread_num();
        printf("OpenMP thread i%d from %d threads \n",
theadNum, nThreads);
    }
```

```
    }  
    return 0;  
}
```

Для запуска программ необходимо использовать следующий скрипт:

```
#!/bin/bash  
#PBS -N hello_MPI  
#PBS -l walltime=00:01:10  
#PBS -l nodes=2:ppn=8  
#PBS -j oe  
#PBS -A tk  
cd $PBS_O_WORKDIR  
  
module load impi/4  
export I_MPI_DEVICE=rdma  
export I_MPI_DEBUG=0  
export I_MPI_FALLBACK_DEVICE=disable  
mpirun -r ssh -machinefile $PBS_NODEFILE -np $PBS_NP ./mpi_1.out
```