

КОМПИЛЯЦИЯ ПРОГРАММ, НАПИСАННЫХ НА ЯЗЫКЕ СИ, В LINUX (в лабораториях Кафедры вычислительных систем СибГУТИ)

В методическом пособии рассматривается порядок разработки программ, включающий:

- подготовку текстового файла с исходным кодом программы;
- компиляцию исходной программы;
- запуск исполняемого файла.

1. Общие сведения.

Прежде всего, следует сказать, что иллюстративный материал основывается на стандартных настройках интерфейса для студентов, выполняющих лабораторные работы в лабораториях Кафедры вычислительных систем Сибирского государственного университета телекоммуникаций и информатики. Интерфейс базируется на оконном менеджере IceWM и после входа в систему имеет вид показанный на рисунке 1. Если Вы используете другой оконный менеджер, или Вы изменили настройки менеджера IceWM, то иллюстративный материал не будет соответствовать тому, что Вы видите на экране. Однако основной принцип выполняемых действий останется неизменным.

Все команды, о которых будет идти речь в методических указаниях, выполняются в терминале (командной строке), вид которого показан на рисунке 2. Запустить терминал можно либо, выбрав соответствующий пункт меню IceWM, либо наведя курсор на значок монитора, расположенного в левом нижнем углу экрана, и щелкнув по нему левой кнопкой мышки. Одновременно может быть запущено несколько терминалов. Переключение между ними осуществляется нажатием клавиш ALT+TAB.

2. Этап подготовки исходного текста программы.

Исходный текст программы должен быть записан в текстовый файл. Для этого используются различные текстовые редакторы, которые могут быть самыми обычными или специально предназначены для написания программ. Здесь будет показано, как пользоваться редактором MCEdit, входящего в состав оболочки Midnight Commander.

Для запуска оболочки Midnight Commander необходимо выполнить команду *mc*, как показано на рисунке 3 (т.е. необходимо набрать *mc* и нажать Enter). После запуска Вы увидите на экране содержимое Вашего домашнего каталога (рисунок 4).

Для создания нового файла надо нажать комбинацию клавиш SHIFT+F4, после чего запустится редактор MCEdit (см. рисунок 5), где Вы можете ввести текст программы (рисунок 6). Для сохранения файла необходимо нажать клавишу F2, после чего будет предложено ввести имя файла (например, *first_program.c*) и нажать Enter. Редактор переспросит Вас о Вашем намерении сохранить файл. После того, как Вы первый раз сохранили файл и задали ему имя, последующие нажатия на клавишу F2 будут сопровождаться только под-

тверждением Вашего намерения сохранить файл. Для подтверждения сохранения необходимо нажать Enter.

Для того, чтобы открыть уже имеющийся файл для редактирования необходимо его выделить с помощью клавиш управления курсором и нажать F4. Сохранение файла производится аналогичным образом.

Чтобы выйти из редактора надо нажать клавишу F10 или два раза клавишу ESC.

3. Этап компиляции программы.

После того, как исходный текст программы подготовлен его необходимо скомпилировать – т.е. преобразовать из текстового вида в формат, понятный для ЭВМ. Для этого надо запустить компилятор, указав ему, где располагается файл и исходным кодом и куда нужно записать файл результат. Команду можно ввести в командной строке Midnight Commander-a (располагающейся под панелями), либо скрыть панели (нажав клавиши CTRL+O) и ввести команду там.

Команда запуска компилятора имеет вид:

gcc имя_файла.c -o имя_исполняемого_файла

где *имя_файла.c* – это имя файла, содержащего исходный текст программы, *имя_исполняемого_файла* – имя файла, куда нужно записать результат компиляции.

Если компиляция прошла удачно, то в списке файлов появится указанный исполняемый файл. В оболочке Midnight Commander такой файл будет выделен цветом.

Чтобы проверить, что компиляция прошла успешно, или посмотреть, почему она не прошла, необходимо скрыть панели оболочки Midnight Commander-a, для чего необходимо нажать клавиши CTRL+O. После этого на экране можно будет увидеть сообщения компилятора.

Для того, чтобы задать режим работы компилятора в командной строке можно задать дополнительные опции, например:

-Wall – показывать все сообщения;

-pedantic – строго проверять исходный код на соответствие стандарту языка Си (без использования умолчаний);

-ansi – проверять строгое соответствие текста программы стандарту ANSI.

-lm – подключить библиотеку математических функций.

4. Этап запуска исполняемой программы.

Для запуска исполняемого файла необходимо его выделить и нажать клавишу Enter, либо ввести в командной строке ./имя_исполняемого_файла и нажать Enter.



Рисунок 1. Вид интерфейса после входа в систему

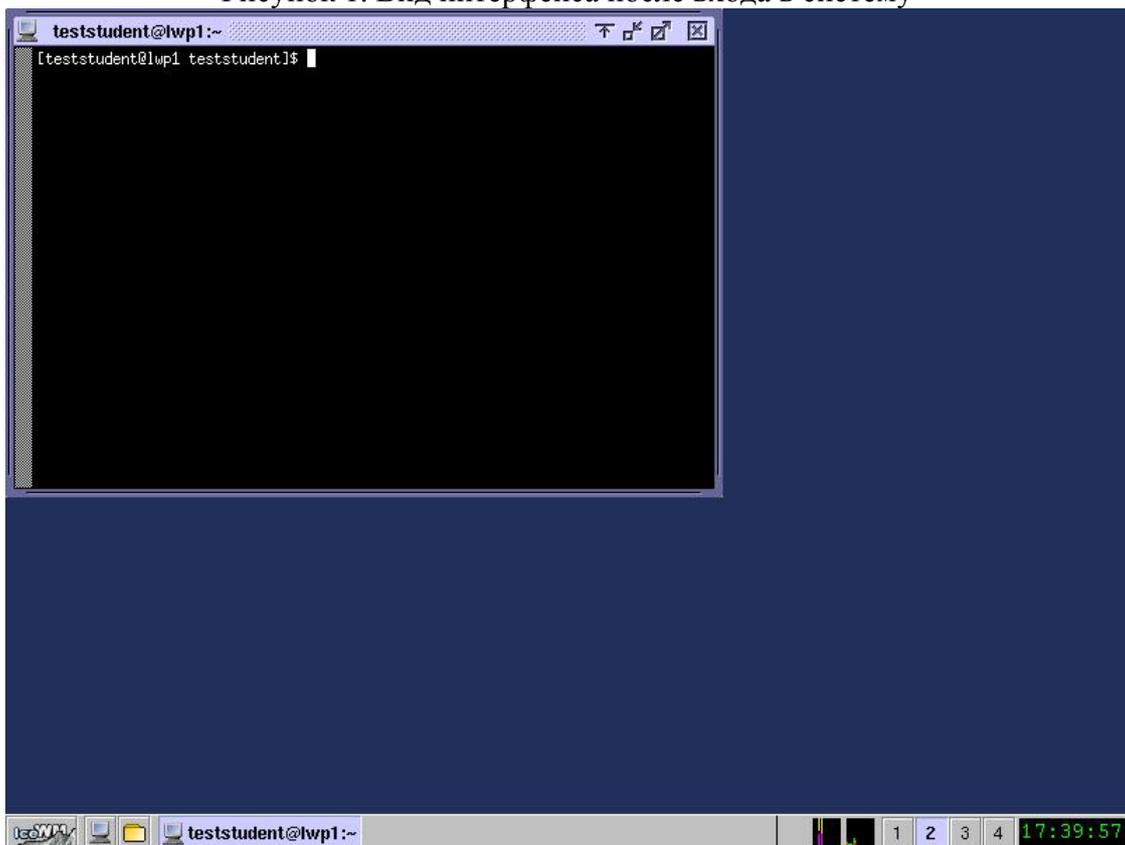


Рисунок 2. Вид экрана после запуска терминала

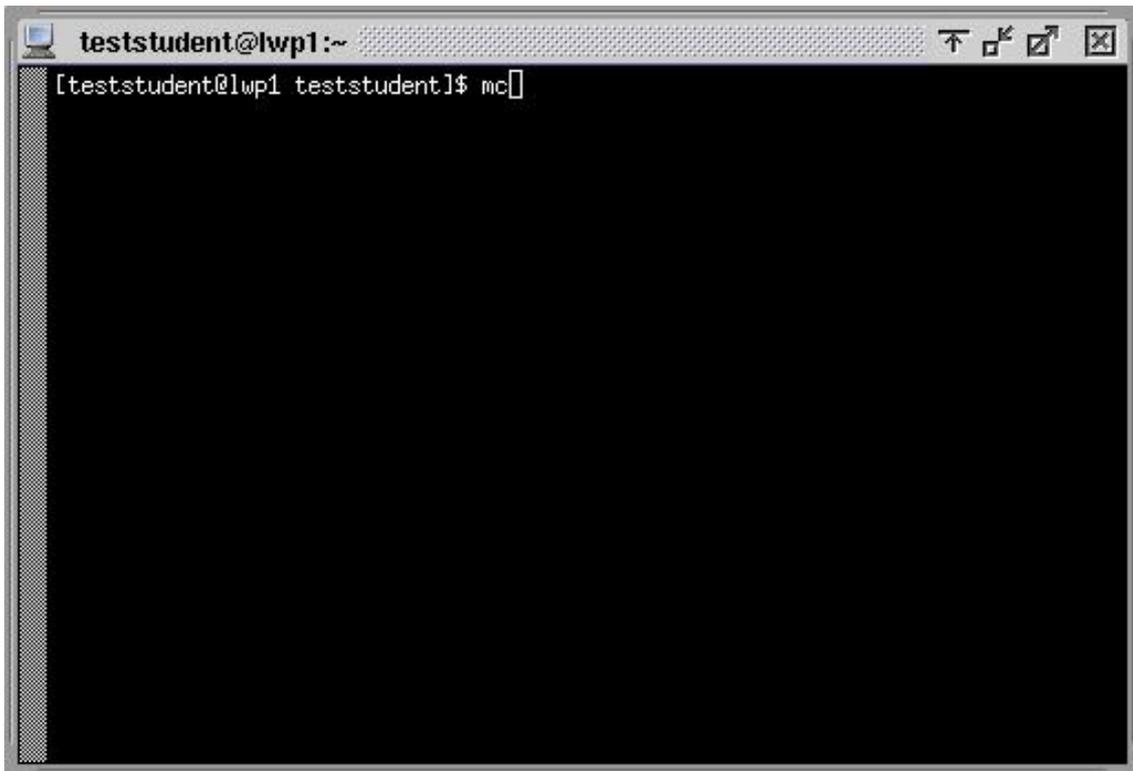


Рисунок 3. Запуск оболочки Midnight Commander

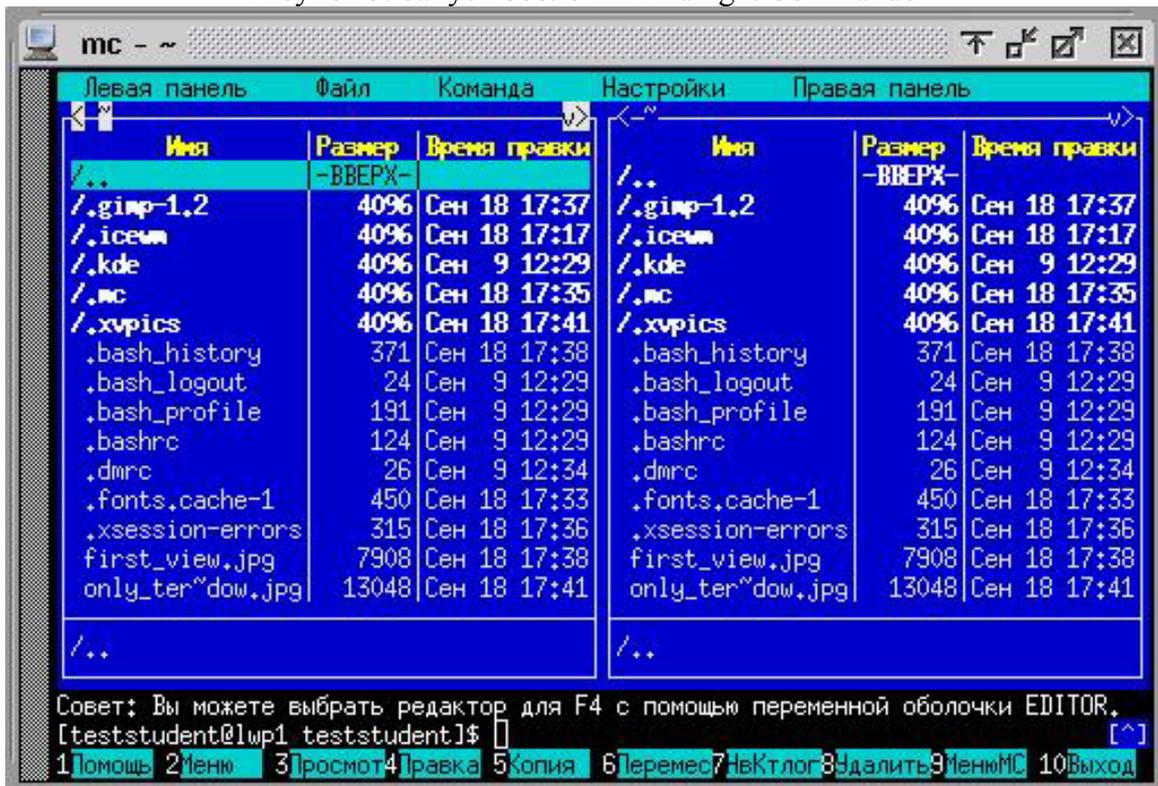


Рисунок 4. Терминал с запущенной программой Midnight Commander

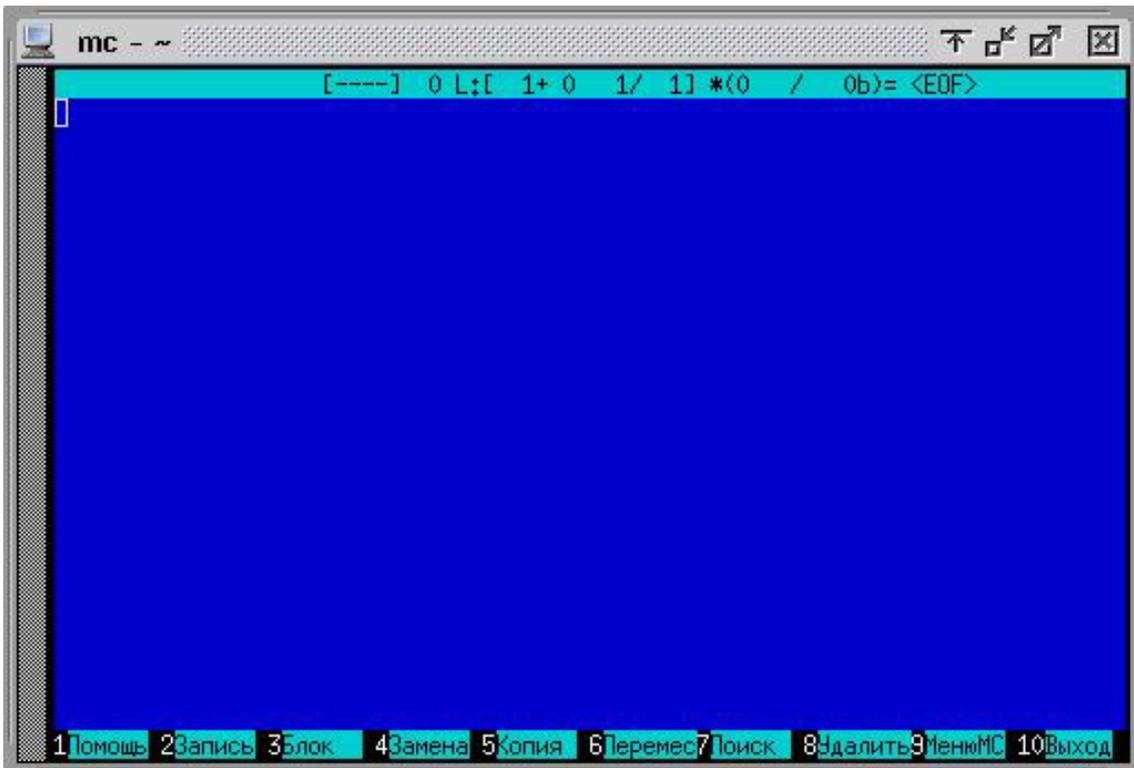


Рисунок 5. Окно терминала с запущенным редактором MCEdit

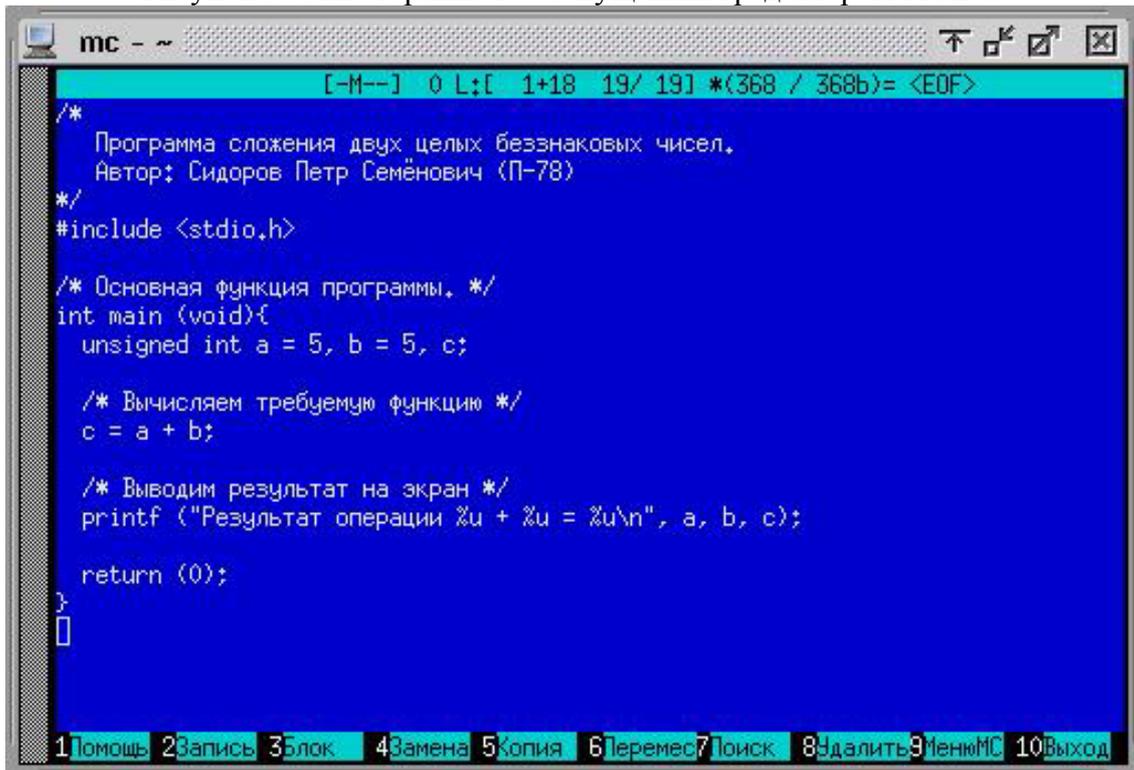


Рисунок 6. Окно терминала после запуска MCEdit и ввода текста программы

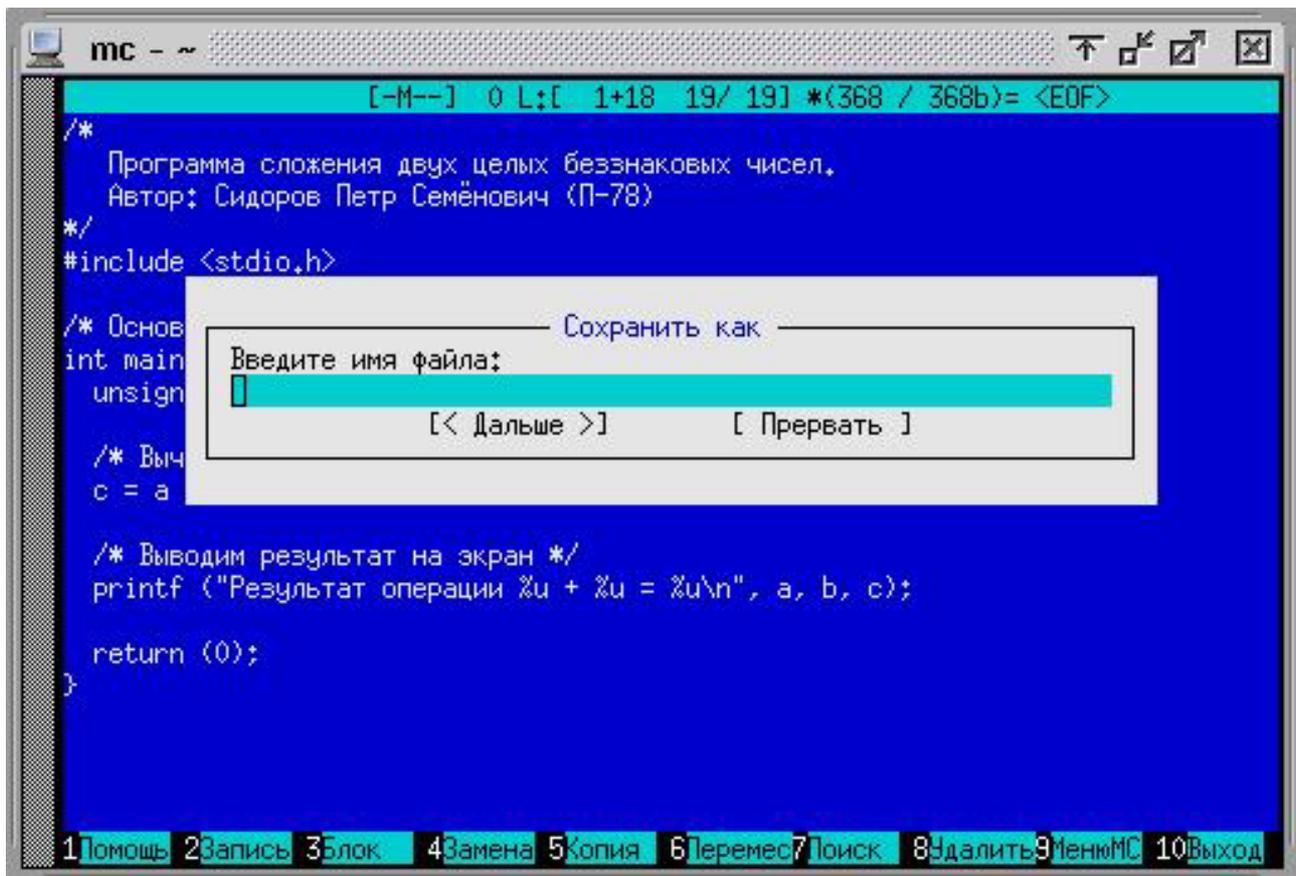


Рисунок 7. Окно редактора MCEdit после нажатия F2

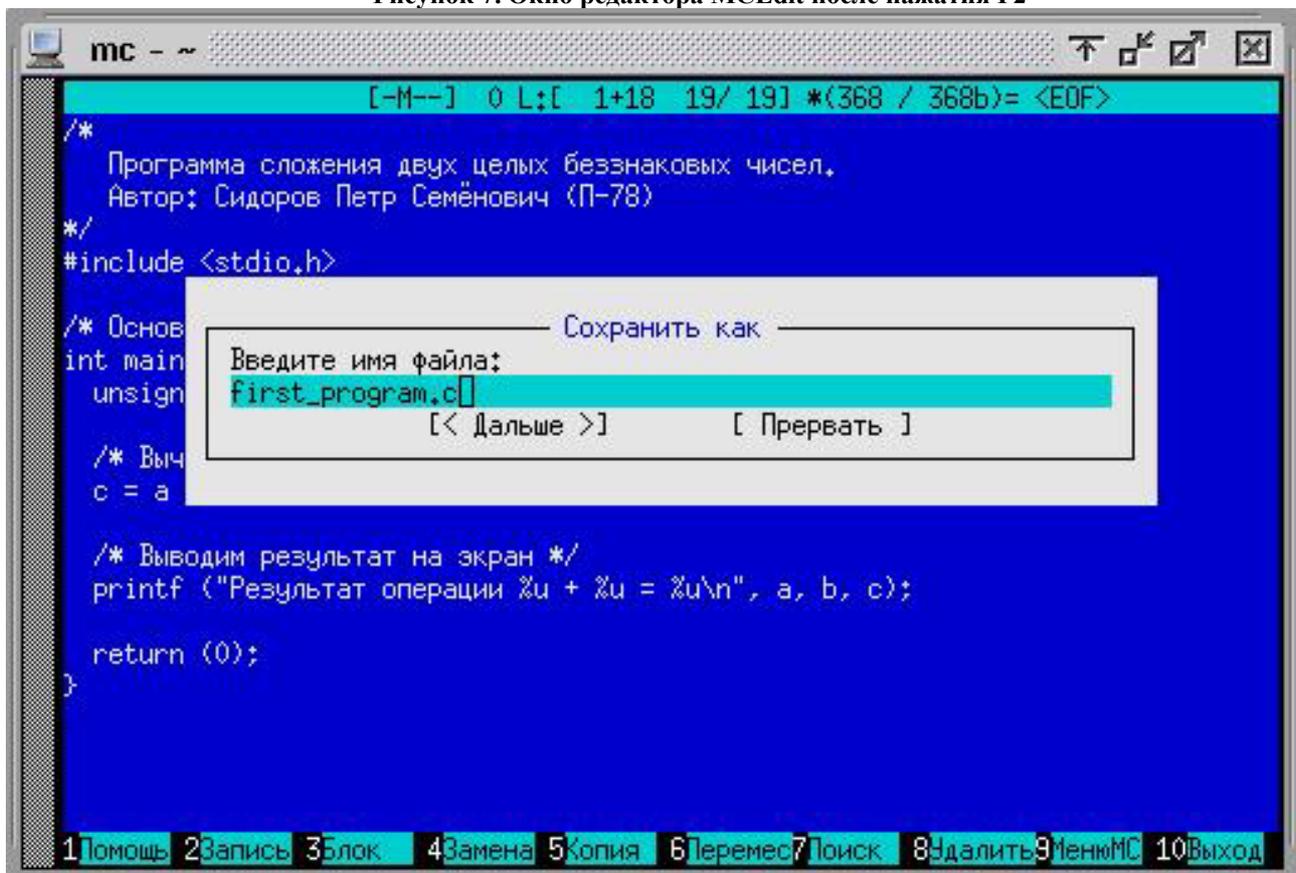


Рисунок 8. Окно редактора MCEdit после ввода имени файла

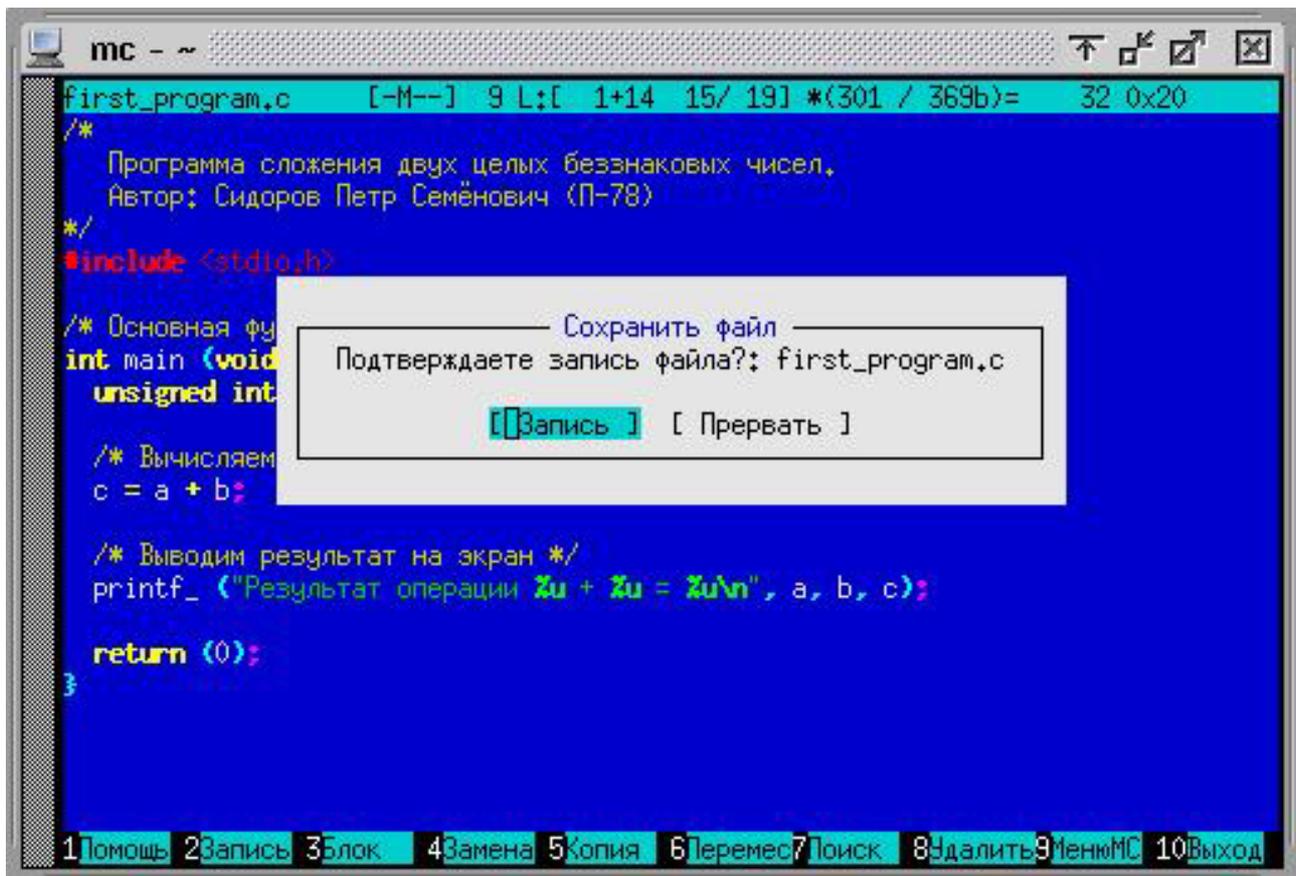


Рисунок 9. Окно редактора после ввода имени и нажатия Enter

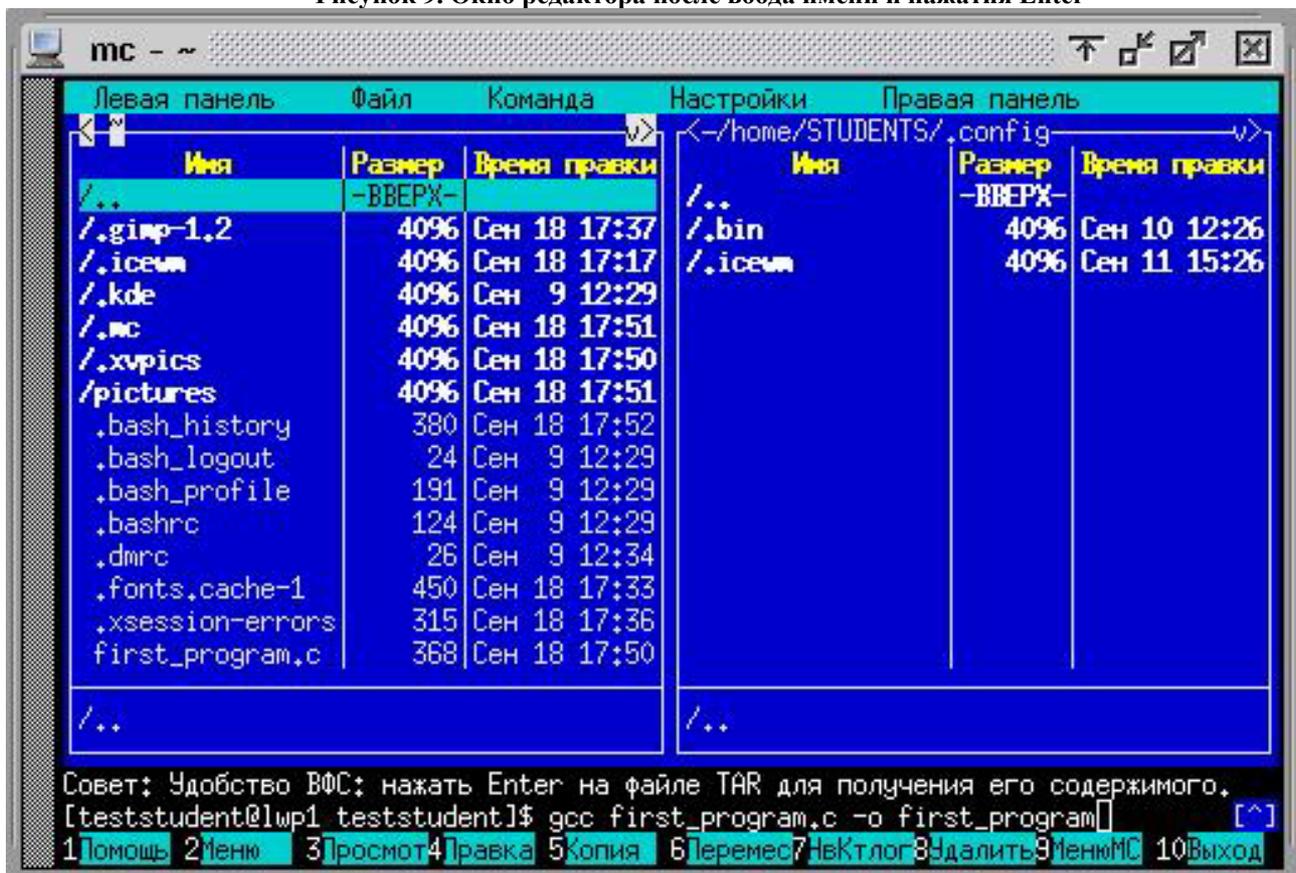


Рисунок 10. Ввод команды по запуску компилятора

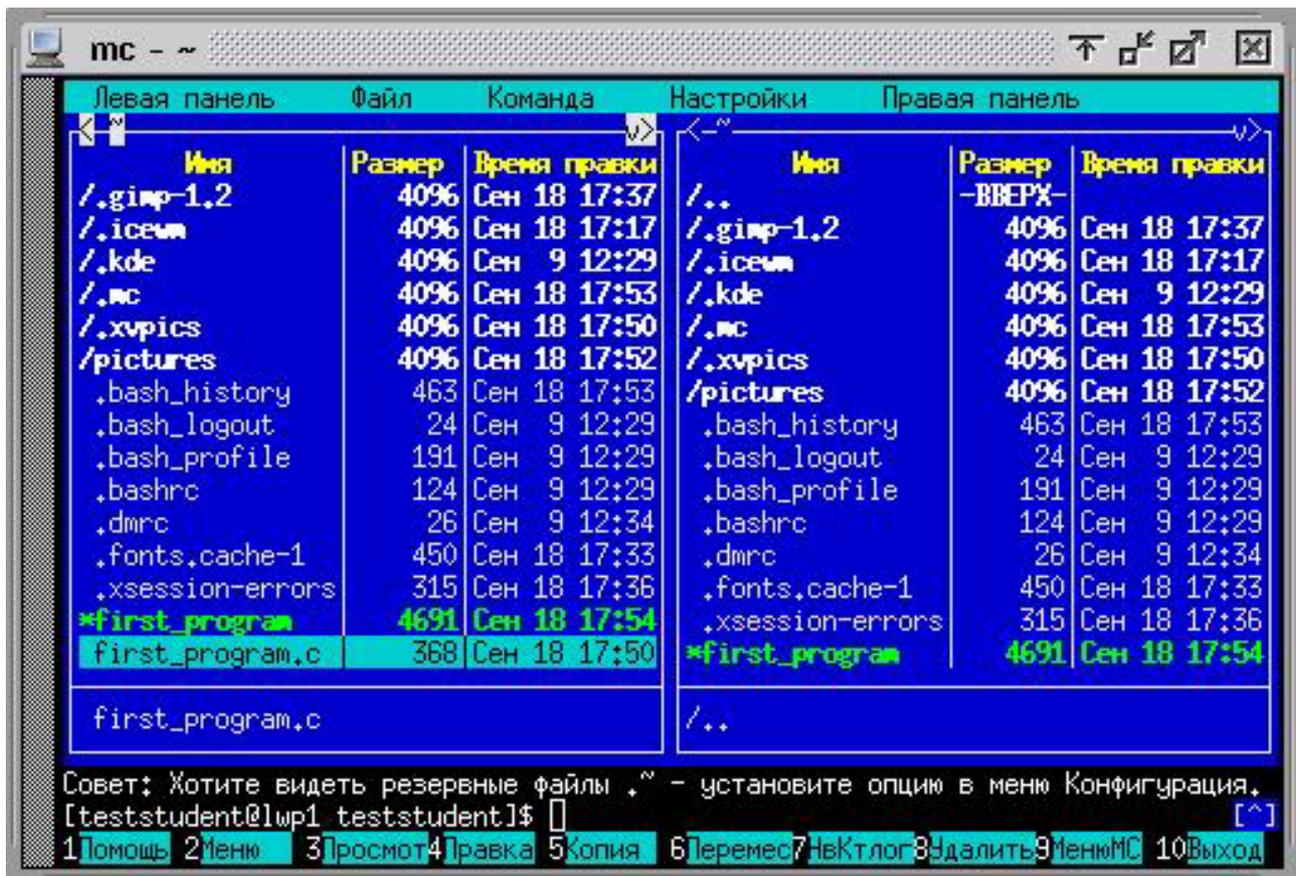


Рисунок 11. Вид терминала после выполнения компилятора

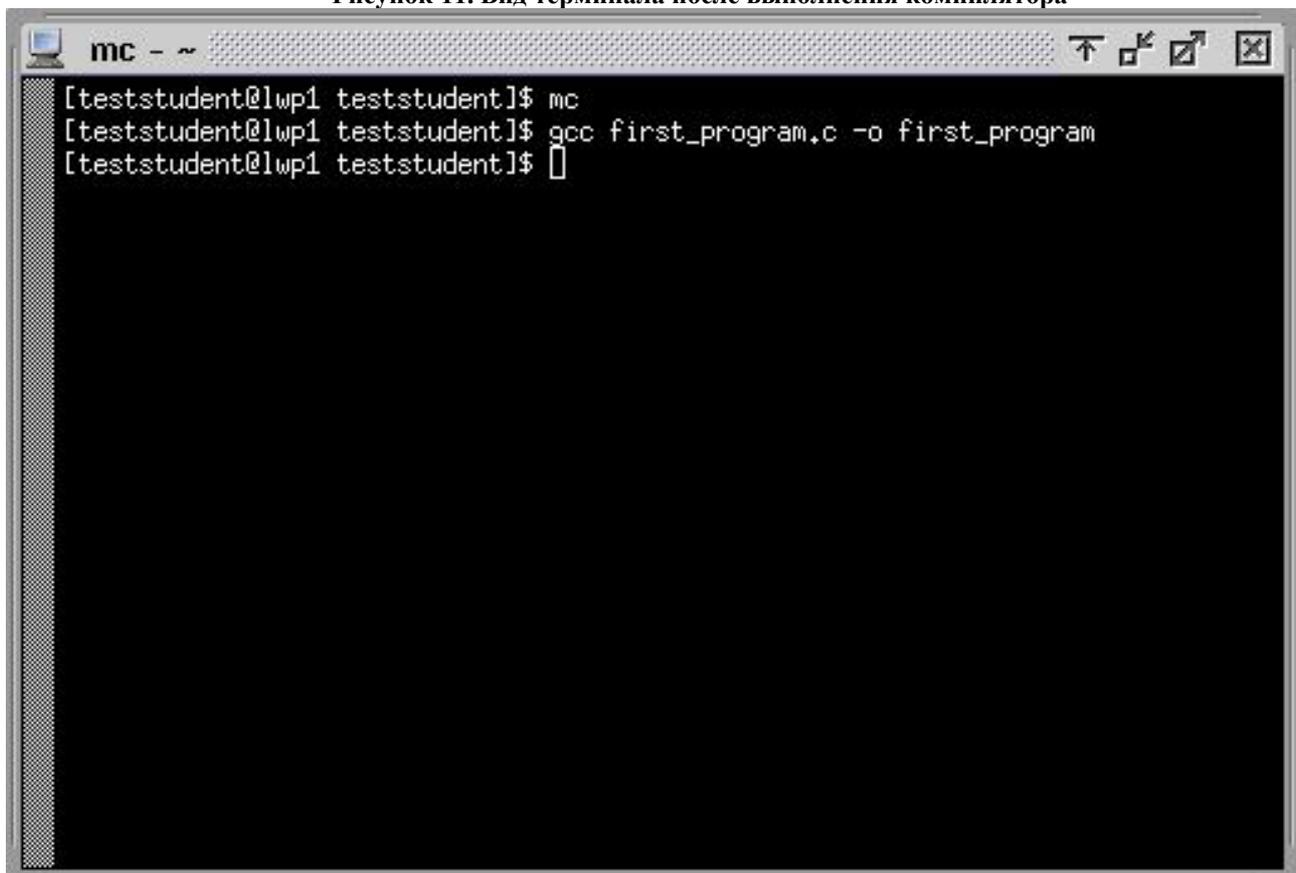
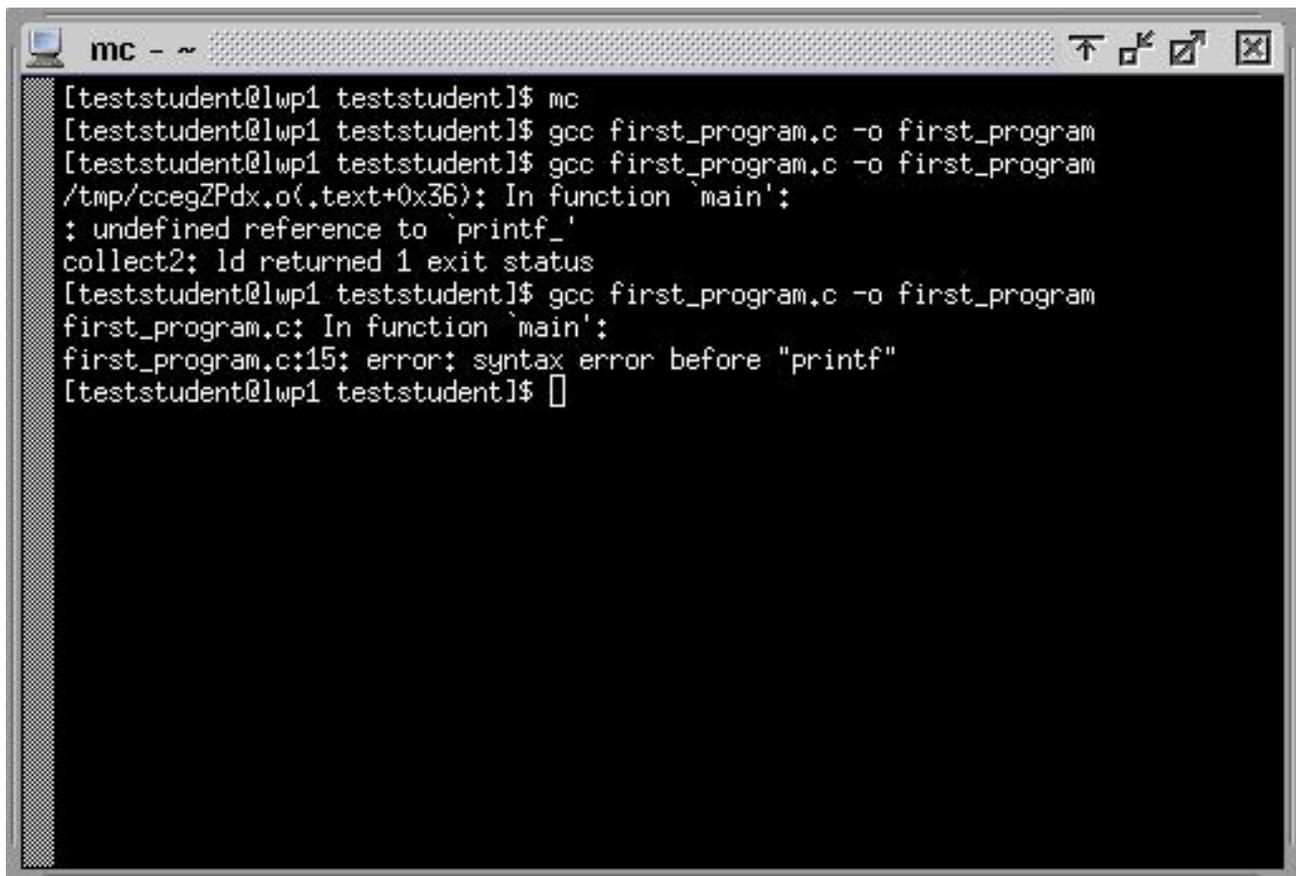
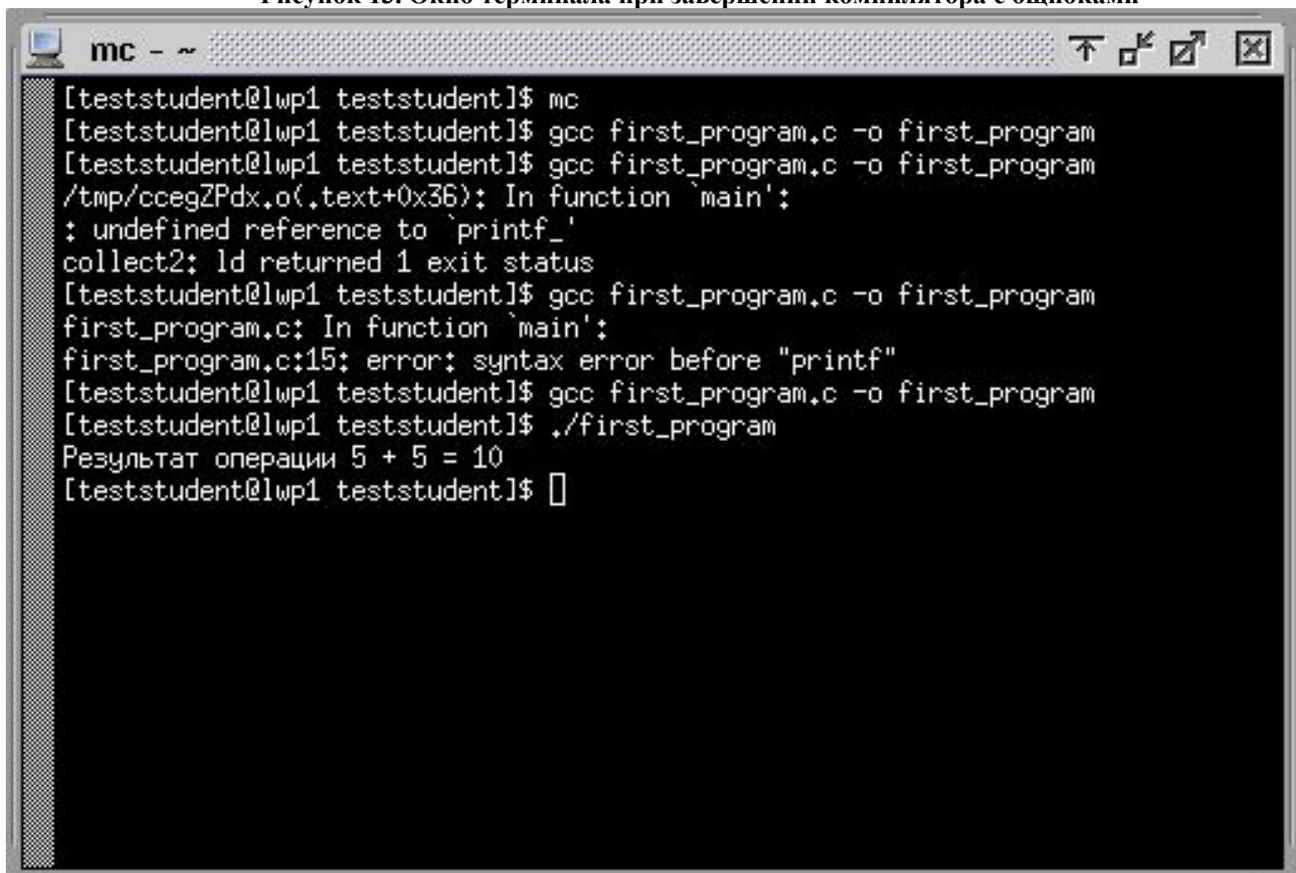


Рисунок 12. Окно терминала после скрытия панелей оболочки



```
mc - ~
[teststudent@lwp1 teststudent]$ mc
[teststudent@lwp1 teststudent]$ gcc first_program.c -o first_program
[teststudent@lwp1 teststudent]$ gcc first_program.c -o first_program
/tmp/ccgZPdx.o(.text+0x36): In function `main':
: undefined reference to `printf_'
collect2: ld returned 1 exit status
[teststudent@lwp1 teststudent]$ gcc first_program.c -o first_program
first_program.c: In function `main':
first_program.c:15: error: syntax error before "printf"
[teststudent@lwp1 teststudent]$
```

Рисунок 13. Окно терминала при завершении компилятора с ошибками



```
mc - ~
[teststudent@lwp1 teststudent]$ mc
[teststudent@lwp1 teststudent]$ gcc first_program.c -o first_program
[teststudent@lwp1 teststudent]$ gcc first_program.c -o first_program
/tmp/ccgZPdx.o(.text+0x36): In function `main':
: undefined reference to `printf_'
collect2: ld returned 1 exit status
[teststudent@lwp1 teststudent]$ gcc first_program.c -o first_program
first_program.c: In function `main':
first_program.c:15: error: syntax error before "printf"
[teststudent@lwp1 teststudent]$ gcc first_program.c -o first_program
[teststudent@lwp1 teststudent]$ ./first_program
Результат операции 5 + 5 = 10
[teststudent@lwp1 teststudent]$
```

Рисунок 14. Окно терминала после запуска программы и скрытия панелей