

# Специальные параметры и переменные оболочки

## Специальные параметры

параметр	символ	описание расширения
*	звездочка	<p>Позиционные параметры, начиная с первого. При использовании внутри двойных кавычек (см. Кавычки), например "\$*" , он распространяется на все позиционные параметры <i>как одно слово</i>, разделенное первым символом IFS переменной (пробел в этом примере): "\$1 \$2 \$3 \$4" .</p> <p>Если IFS значение не задано, используемый разделитель всегда будет пробелом, если IFS равно NULL, разделителем будет ничто, что эффективно объединяет все позиционные параметры без какого-либо разделителя.</p> <p>При использовании без кавычек он будет просто расширяться до строк, одну за другой, не сохраняя границ слов (т. Е. Разделение слов снова разделит текст, если он содержит IFS символы.</p> <p>См. Также статью о скриптах об обработке позиционных параметров.</p>
@	при входе	<p>Позиционные параметры, начиная с первого. При использовании внутри двойных цитат (см. Кавычки), например "\$@" , он расширяет все позиционные параметры <i>как отдельные слова</i>: "\$1" "\$2" "\$3" "\$4"</p> <p>Без двойных ссылок поведение аналогично поведению * без двойных ссылок.</p> <p>См. Также статью о скриптах об обработке позиционных параметров.</p>
#	хэш-метка	<p>Количество позиционных параметров (десятичных)</p> <p>См. Также статью о скриптах об обработке позиционных параметров.</p>
?	знак вопроса	<p>Состояние самого последнего выполненного конвейера переднего плана (код выхода / возврата)</p>

параметр	символ	описание расширения
-	dash	Текущие флаги параметров устанавливаются самой оболочкой, при вызове или с помощью встроенной команды <code>set</code> . Это просто набор символов, например <code>himB</code> , для <code>h</code> , <code>i</code> , <code>m</code> и <code>B</code> .
\$	знак доллара	Идентификатор процесса (PID) оболочки. В явной подоболочке он расширяется до PID текущей "основной оболочки", а не подоболочки. Это отличается от <code>\$BASHPID</code> !
!	восклицательный знак	Идентификатор процесса (PID) самого последнего выполненного фонового конвейера (например, начинался с <code>command &amp;</code> )
0	ноль	Имя оболочки или сценария оболочки (имя файла). Устанавливается самой оболочкой. Если Bash запускается с именем файла для выполнения (script), ему присваивается это имя файла. Если начато с <code>-c &lt;CMDLINE&gt;</code> параметра (командная строка, указанная в качестве аргумента), то <code>\$0</code> это будет первый аргумент после заданного <code>&lt;CMDLINE&gt;</code> . В противном случае он устанавливается в строку, указанную при вызове <code>for argv[0]</code> . В отличие от распространенного мнения, <code>\$0</code> не является позиционным параметром.
_	подчеркивание	Своего рода универсальный параметр. Непосредственно после вызова оболочки для него устанавливается имя файла, используемое для вызова Bash, или абсолютный или относительный путь к скрипту, точно так же, как <code>\$0</code> это было бы показано. Впоследствии расширяется до последнего аргумента предыдущей команды. Помещается в среду при выполнении команд и устанавливается в полное имя пути этих команд. При проверке почты этот параметр содержит имя проверяемого в данный момент почтового файла.

# Переменные оболочки

## BASH

Переменная:	BASH	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ

<b>Устанавливается</b> <b>Bash:</b>	ДА	<b>По умолчанию:</b>	n/a
--	----	----------------------	-----

Расширяется до полного имени файла, используемого для вызова текущего экземпляра Bash.

## BASHOPTS

<b>Переменная:</b>	BASHOPTS	<b>Поскольку:</b>	4.1- альфа
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	ДА
<b>Устанавливается</b> <b>Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Список включенных параметров оболочки, разделенных двоеточием.

Каждое слово в списке является допустимым аргументом для `-s` опции встроенной команды `shopt`. Параметры, отображаемые в `BASHOPTS`, - это те, о которых сообщается как `on by shopt`. Если эта переменная находится в среде при запуске Bash, каждая опция оболочки в списке будет включена перед чтением любых файлов запуска.

Пример содержимого:

```
cmdhist:expand_aliases:extquote:force_ignore:hostcomplete:interactiv  
e_comments:progcomp:promptvars:sourcepath
```

Эта переменная доступна только для чтения.

## BASHPID

<b>Переменная:</b>	BASHPID	<b>Поскольку:</b>	4.0- альфа
<b>Тип:</b>	целочисленная переменная	<b>Доступно только для чтения:</b>	ДА
<b>Устанавливается</b> <b>Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Всегда расширяется до идентификатора процесса текущего процесса Bash. Это отличается от специального параметра `$` при определенных обстоятельствах, таких как подоболочки, которые не требуют повторной инициализации Bash.

## BASH\_ALIASES

<b>Переменная:</b>	BASH_ALIASES	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	ассоциативный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Переменная ассоциативного массива, члены которой соответствуют внутреннему списку псевдонимов, поддерживаемому встроенным псевдонимом. Элементы, добавленные в этот массив, отображаются в списке псевдонимов; отключение элементов массива приводит к удалению псевдонимов из списка псевдонимов.

Ассоциативный ключ - это имя псевдонима, используемого со встроенной командой alias.

## BASH\_ARGC

<b>Переменная:</b>	BASH_ARGC	<b>Поскольку:</b>	3.0
<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	только в режиме расширенной отладки	<b>По умолчанию:</b>	n/a

Переменная массива, значениями которой являются количество параметров в каждом кадре текущего стека вызовов выполнения Bash.

Количество параметров текущей подпрограммы (функции оболочки или скрипта, выполняемых с помощью встроенной команды "." или "source") находится в верхней части стека. Когда выполняется подпрограмма, количество переданных параметров передается BASH\_ARGC .

## BASH\_ARGV

<b>Переменная:</b>	BASH_ARGV	<b>Поскольку:</b>	3.0
<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	только в режиме расширенной отладки	<b>По умолчанию:</b>	n/a

Переменная массива, содержащая все параметры в текущем стеке вызовов выполнения Bash.

Последний параметр последнего вызова подпрограммы находится в верхней части стека; первый параметр начального вызова находится внизу. При выполнении подпрограммы вводятся предоставленные параметры BASH\_ARGV .

## BASH\_ARGV0

Переменная:	BASH_ARGV0	Поскольку:	5.0-альфа
Тип:	строка	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	то же, что \$0

Расширяется до имени оболочки или сценария оболочки - как это делает специальный параметр \$0 . Присваивание BASH\_ARGV0 вызывает присвоение значения \$0 .

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии сбросить.

## BASH\_CMDS

Переменная:	BASH_CMDS	Поскольку:	неизвестно
Тип:	ассоциативный массив	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Переменная ассоциативного массива, члены которой соответствуют внутренней хэш-таблице команд, поддерживаемой встроенной командой hash. Элементы, добавленные в этот массив, отображаются в хэш-таблице; отключение элементов массива приводит к удалению команд из хэш-таблицы.

Ассоциативный ключ - это имя команды, используемой со встроенной командой hash.

## BASH\_COMMAND

Переменная:	BASH_COMMAND	Поскольку:	3.0
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Команда, выполняемая в данный момент или собирающаяся быть выполненной, если только оболочка не выполняет команду в результате ловушки, и в этом случае это команда, выполняемая во время ловушки.

## BASH\_COMPAT

Переменная:	BASH_COMPAT	Поскольку:	4.3- альфа
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Значение используется для установки уровня совместимости оболочки. Значение может быть десятичным числом (например, 4.2 ) или целым числом (например, 42 ), соответствующим желаемому уровню совместимости. Если BASH\_COMPAT значение не задано или установлено в пустую строку, уровень совместимости устанавливается по умолчанию для текущей версии. Если BASH\_COMPAT установлено значение, которое не является одним из допустимых уровней совместимости, оболочка выводит сообщение об ошибке и устанавливает уровень совместимости по умолчанию для текущей версии. Допустимые уровни совместимости соответствуют параметрам совместимости, принятым встроенным shopt. Текущая версия также является допустимым значением.

## BASH\_EXECUTION\_STRING

Переменная:	BASH_EXECUTION_STRING	Поскольку:	3.0
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Аргумент команды для параметра -с вызова.

## BASH\_LINENO

Переменная:	BASH_LINENO	Поскольку:	3.0
Тип:	целочисленный индексированный массив	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Переменная массива, членами которой являются номера строк в исходных файлах, соответствующие каждому члену FUNCNAME .

`${BASH_LINENO[$i]}` это номер строки в исходном файле, где `${FUNCNAME[$iFP]}` был вызван. Соответствующее имя исходного файла `${BASH_SOURCE[$i]}` .

Используется LINENO для получения текущего номера строки.

## BASH\_REMATCH

Переменная:	BASH_REMATCH	Поскольку:	3.0
Тип:	целочисленный индексированный массив	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Переменная массива, члены которой назначаются `=~` двоичным оператором `[]` условной команде.

Элемент с индексом `0` - это часть строки, соответствующая всему регулярному выражению. Элемент с индексом `n` - это часть строки, соответствующая `n`-му подвыражению, заключенному в скобки.

До версии Bash 5.1-alpha эта переменная была доступна только для чтения.

## BASH\_SOURCE

Переменная:	BASH_SOURCE	Поскольку:	3.0
Тип:	целочисленный индексированный массив	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Переменная массива, членами которой являются исходные имена файлов, соответствующие элементам в переменной `FUNCNAME` массива.

## BASH\_SUBSHELL

Переменная:	BASH_SUBSHELL	Поскольку:	3.0
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Увеличивается на единицу при каждом создании подоболочки или подоболочечной среды. Начальное значение равно `0`.

## BASH\_VERSIONINFO

Переменная:	BASH_VERSIONINFO	Поскольку:	2.0
-------------	------------------	------------	-----

<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	ДА
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Переменная массива, доступная только для чтения, члены которой содержат информацию о версии для этого экземпляра Bash. Значения, присвоенные элементам массива, следующие:

BASH_VERSION[0]	Основной номер версии (релиз)
BASH_VERSION[1]	Младший номер версии (версия)
BASH_VERSION[2]	Уровень исправлений
BASH_VERSION[3]	Версия сборки
BASH_VERSION[4]	Статус выпуска (например, beta1)
BASH_VERSION[5]	Значение MACHTYPE

## BASH\_VERSION

<b>Переменная:</b>	BASH_VERSION	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Расширяется до строки, описывающей версию этого экземпляра Bash.

Начиная с Bash 2.0, он включает в себя "статус выпуска" оболочки (альфа [N], бета [N], релиз).

## CHILD\_MAX

<b>Переменная:</b>	CHILD_MAX	<b>Поскольку:</b>	4.3- альфа
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Задайте количество завершенных значений дочерних статусов, которые должна запомнить оболочка. Bash не позволит уменьшить это значение ниже минимального значения, установленного POSIX, и существует максимальное значение (в настоящее



время 8192), которое оно не может превышать. Минимальное значение зависит от системы.

## COMP\_CWORD

<b>Переменная:</b>	COMP_CWORD	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	только для программируемых средств завершения	<b>По умолчанию:</b>	n/a

Указатель на COMP\_WORDS слово, содержащее текущую позицию курсора.

## COMP\_KEY

<b>Переменная:</b>	COMP_KEY	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	только для программируемых средств завершения	<b>По умолчанию:</b>	n/a

Ключ (или последний ключ последовательности ключей), используемый для вызова текущей функции завершения.

## COMP\_LINE

<b>Переменная:</b>	COMP_LINE	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	только для программируемых средств завершения	<b>По умолчанию:</b>	n/a

Текущая командная строка.

## COMP\_POINT

<b>Переменная:</b>	COMP_POINT	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ

Устанавливается только для программируемых Bash: средств завершения

По умолчанию: n/a


Индекс текущего положения курсора относительно начала текущей команды. Если текущая позиция курсора находится в конце текущей команды, значение этой переменной равно `${#COMP_LINE}` .

## COMP\_TYPE

Переменная:	COMP_TYRET	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	только для программируемых средств завершения	По умолчанию:	n/a

Устанавливается в целочисленное значение, соответствующее типу попытки завершения, которая вызвала вызов функции завершения:

ТАВ	обычное завершение
?	список завершений после последовательных вкладок
!	список альтернатив для частичного завершения word
@	чтобы перечислить дополнения, если слово не является неизменным
%	для завершения меню

 где целочисленные значения?

## COMP\_WORDBREAKS

Переменная:	COMP_WORDBREAKS	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Сообщает набор символов, которые библиотека readline обрабатывает как разделители слов при выполнении завершения слова.

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## COMP\_WORDS

<b>Переменная:</b>	COMP_WORDS	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	только для программируемых средств завершения	<b>По умолчанию:</b>	n/a

Переменная массива, состоящая из отдельных слов в текущей командной строке. Строка разбивается на слова так, как ее разделил бы readline, используя COMP\_WORDS\_BREAKS , как описано выше.

## COPROC

<b>Переменная:</b>	COPROC	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Переменная массива, созданная для хранения файловых дескрипторов для вывода и ввода в неназванный сопроцесс.

## DIRSTACK

<b>Переменная:</b>	DIRSTACK	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Переменная массива, содержащая текущее содержимое стека каталогов.

Каталоги отображаются в стеке в том порядке, в котором они отображаются встроенными каталогами. Присвоение членам этой переменной массива может использоваться для изменения каталогов, уже находящихся в стеке, но встроенные функции pushd и popd должны использоваться для добавления и удаления каталогов.

Присвоение этой переменной не изменит текущий каталог.

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## ЭПОХАЛЬНОЕ ВРЕМЯ

Переменная:	EPOCHREALTIME	Поскольку:	5.0- альфа
Тип:	целочисленная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Расширяется до количества секунд, прошедших с момента Unix epoch, в виде значения с плавающей запятой с детализацией в микросекунды.

Присвоения этому параметру игнорируются. Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## СЕКУНДЫ ЭПОХИ

Переменная:	EPOCHSECONDS	Поскольку:	5.0- альфа
Тип:	целочисленная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Расширяется до количества секунд, прошедших с момента запуска Unix.

Присвоения этому параметру игнорируются. Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## EUID

Переменная:	EUID	Поскольку:	неизвестно
Тип:	целочисленная переменная	Доступно только для чтения:	ДА
Устанавливается Bash:	ДА	По умолчанию:	n/a

Расширяется до действительного идентификатора текущего пользователя, инициализируемого при запуске оболочки.

⚠ Не полагайтесь на эту переменную, когда речь идет о безопасности.

## FUNCNAME

Переменная:	FUNCNAME	Поскольку:	2.04
-------------	----------	------------	------

<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	только внутри функций оболочки	<b>По умолчанию:</b>	n/a

Переменная массива, содержащая имена всех функций оболочки, находящихся в данный момент в стеке вызовов выполнения.

Элемент с индексом 0 - это имя любой выполняемой в данный момент функции оболочки. Самый нижний элемент (тот, у которого самый высокий индекс) - "main".

Эта переменная может использоваться с BASH\_LINENO и BASH\_SOURCE : каждый элемент FUNCNAME имеет соответствующие элементы в BASH\_LINENO и BASH\_SOURCE для описания стека вызовов. Например, \${FUNCNAME[\$i]} был вызван из файла \${BASH\_SOURCE[\$i+1]} с номером строки \${BASH\_LINENO[\$i]} .

Встроенная команда вызывающего абонента отображает текущий стек вызовов, используя эту информацию.

Эта переменная существует только тогда, когда выполняется функция оболочки.

Присвоение этому параметру не имеет никакого эффекта и возвращает статус ошибки.

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## Группы

<b>Переменная:</b>	GROUPS	<b>Поскольку:</b>	2.01
<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Переменная массива, содержащая список групп, членом которых является текущий пользователь.

Присвоение этому параметру не имеет никакого эффекта и возвращает статус ошибки.

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## HISTCMD

<b>Переменная:</b>	HISTCMD	<b>Поскольку:</b>	1.14.0
<b>Тип:</b>	целочисленная переменная	<b>Доступно только для чтения:</b>	НЕТ

<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a
----------------------------------	----	----------------------	-----

Расширяется до номера истории (индекса в списке истории) текущей команды.

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## ИМЯ хоста

<b>Переменная:</b>	HOSTNAME	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Автоматически устанавливается на имя текущего хоста.

## ТИП хоста

<b>Переменная:</b>	HOSTTYPE	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	зависит от системы

Автоматически устанавливается в строку, которая однозначно описывает тип компьютера, на котором выполняется Bash.

Пример содержимого:

x86\_64

## LINENO

<b>Переменная:</b>	LINENO	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	целочисленная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Каждый раз, когда на этот параметр ссылаются, оболочка заменяет десятичное число, представляющее текущий порядковый номер строки (начиная с 1) в скрипте или функции.

Если их нет в скрипте или функции, подставляемое значение не гарантированно будет значимым.

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## MASHTYPE

Переменная:	MASHTYPE	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	зависит от системы

Автоматически устанавливается в строку, которая полностью описывает тип системы, на которой выполняется Bash, в стандартном формате GNU "cpu-company-system".

Пример содержимого:

```
x86_64-неизвестно-linux-gnu
```

## ФАЙЛ КАРТЫ

Переменная:	MAPFILE	Поскольку:	неизвестно
Тип:	целочисленный индексированный массив	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Переменная массива, созданная для хранения текста, считываемого встроенной командой mapfile, когда имя переменной не указано.

## OLDPWD

Переменная:	OLDPWD	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Предыдущий рабочий каталог, установленный командой cd.

## OPTARG

Переменная:	OPTARG	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Значение последнего параметра option, обрабатываемого встроенной командой getopts.

## OPTIND

Переменная:	OPTIND	Поскольку:	неизвестно
Тип:	целочисленная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Индекс следующего аргумента, который будет обработан встроенной командой getopts.

## OSTYPE

Переменная:	OSTYPE	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	зависит от системы

Автоматически устанавливается в строку, описывающую операционную систему, в которой выполняется Bash.

Пример содержимого:

```
linux-gnu
```

## PIPESTATUS



<b>Переменная:</b>	PIPESTATUS	<b>Поскольку:</b>	2.0
<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Переменная массива, содержащая список значений статуса завершения процессов в последнем выполненном конвейере переднего плана (который может содержать только одну команду).

## PPID

<b>Переменная:</b>	PPID	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	целочисленная переменная	<b>Доступно только для чтения:</b>	ДА
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Идентификатор процесса родительского процесса оболочки.

## PWD

<b>Переменная:</b>	PWD	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Текущий рабочий каталог, установленный встроенной командой cd.

## Случайный

<b>Переменная:</b>	RANDOM	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	целочисленная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Каждый раз, когда на этот параметр ссылаются, генерируется случайное целое число от 0 до 32767. Последовательность случайных чисел может быть инициализирована путем присвоения значения RANDOM .

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## READLINE\_LINE

Переменная:	READLINE_LINE	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Содержимое буфера строки readline для использования с. bind -x

## READLINE\_POINT

Переменная:	READLINE_POINT	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	n/a

Положение точки вставки в буфере строки readline для использования с. bind -x

## ОТВЕТИТЬ

Переменная:	REPLY	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	только с помощью встроенной команды read	По умолчанию:	n/a

Устанавливается в строку ввода, считываемую встроенной командой read, когда не указаны аргументы, которые называют целевые переменные.

## СЕКУНДЫ

Переменная:	SECONDS	Поскольку:	неизвестно
Тип:	целочисленная переменная	Доступно только для чтения:	НЕТ

<b>Устанавливается</b>	ДА	<b>По умолчанию:</b>	n/a
<b>Bash:</b>			

Каждый раз, когда на этот параметр ссылаются, возвращается количество секунд с момента вызова оболочки. Если значение присваивается СЕКУНДАМ, значение, возвращаемое при последующих ссылках, представляет собой количество секунд с момента присвоения плюс присвоенное значение.

Если этот параметр не установлен, он теряет свои специальные свойства, даже если впоследствии он будет сброшен.

## ШЕЛЛОПТЫ

<b>Переменная:</b>	SHELLOPTS	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	ДА
<b>Устанавливается</b>	ДА	<b>По умолчанию:</b>	n/a
<b>Bash:</b>			

Список включенных параметров оболочки, разделенных двоеточием. Каждое слово в списке является допустимым аргументом для `set` -o параметра встроенной команды `set`. Параметры, отображаемые в `SHELLOPTS`, - это те, о которых сообщается как `on by set -o`.

Если эта переменная находится в среде при запуске Bash, каждая опция оболочки в списке будет включена перед чтением любых файлов запуска.

## SHLVL

<b>Переменная:</b>	SHLVL	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается</b>	ДА	<b>По умолчанию:</b>	n/a
<b>Bash:</b>			

Увеличивается на единицу при каждом запуске экземпляра Bash.

## UID

<b>Переменная:</b>	UID	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	целочисленная переменная	<b>Доступно только для чтения:</b>	ДА

<b>Устанавливается</b> <b>Bash:</b>	ДА	<b>По умолчанию:</b>	n/a
--	----	----------------------	-----

Расширяется до идентификатора текущего пользователя, инициализируемого при запуске оболочки.

⚠ Не полагайтесь на эту переменную, когда речь идет о безопасности.

## BASH\_ENV

<b>Переменная:</b>	BASH_ENV	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается</b> <b>Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Если этот параметр установлен, когда Bash выполняет сценарий оболочки, его значение интерпретируется как имя файла, содержащее команды для инициализации оболочки, как в ~/.bashrc . Значение BASH\_ENV подвергается

- расширение параметров
- подстановка команд
- арифметическое расширение

перед интерпретацией как имя файла.

PATH не используется для поиска результирующего имени файла.

## BASH\_XTRACEFD

<b>Переменная:</b>	BASH_XTRACEFD	<b>Поскольку:</b>	4.1-альфа
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается</b> <b>Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Если задано целое число, соответствующее допустимому файловому дескриптору, Bash запишет вывод трассировки, сгенерированный при set -x включении, в этот файловый дескриптор.

Файловый дескриптор закрывается, если BASH\_XTRACEFD он не установлен или ему присвоено новое значение.

Отключение BASH\_XTRACEFD или присвоение ему пустой строки приводит к тому, что вывод трассировки отправляется со стандартной ошибкой. Обратите внимание, что установка BASH\_XTRACEFD значения 2 (стандартный дескриптор файла ошибки), а затем его отмена приведет к закрытию стандартной ошибки.

# CDPATH

Переменная:	CDPATH	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Путь поиска для встроенной команды `cd`.

Это список каталогов, разделенных двоеточием, в котором оболочка ищет целевые каталоги, указанные `cd` командой.

Пример содержимого:

```
.:~:/usr
```

# СТОЛБЦЫ

Переменная:	COLUMNS	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	на SIGWINCH	По умолчанию:	n/a

Используется составной командой `select` для определения ширины терминала при печати списков выбора. Автоматически устанавливается при получении а `SIGWINCH` .

# ПОНЯТНО

Переменная:	COMPREPLY	Поскольку:	неизвестно
Тип:	целочисленный индексированный массив	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Переменная массива, из которой Bash считывает возможные завершения, сгенерированные функцией оболочки, вызываемой средством программируемого завершения.

# EMACS

<b>Переменная:</b>	EMACS	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Если Bash обнаруживает эту переменную в среде, когда оболочка запускается со значением "t", предполагается, что оболочка выполняется в буфере оболочки Emacs, и редактирование строк отключается.

## ENV

<b>Переменная:</b>	ENV	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Аналогично BASH\_ENV : используется, когда оболочка вызывается в режиме POSIX®.

## FCEDIT

<b>Переменная:</b>	FCEDIT	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Редактор по умолчанию для встроенной команды fc.

## FIGNORE

<b>Переменная:</b>	FIGNORE	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Список суффиксов, разделенных двоеточием, которые следует игнорировать при выполнении завершения имени файла. Имя файла, суффикс которого совпадает с одной из записей в FIGNORE , исключается из списка совпадающих имен файлов.

Пример содержимого:

```
.o:~
```

## FUNCNEST

Переменная:	FUNCNEST	Поскольку:	4.2- альфа
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Если установлено числовое значение больше 0, определяет максимальный уровень вложенности функции. Вызовы функций, превышающие этот уровень вложенности, приведут к прерыванию текущей команды.

Отрицательные значения, 0 или нечисловые присвоения имеют эффект, как если FUNCNEST бы значение было сброшено или пусто: нет элемента управления гнездом

## GLOBIGNORE

Переменная:	GLOBIGNORE	Поскольку:	2.0
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Список шаблонов, разделенных двоеточием, определяющий набор имен файлов, которые будут игнорироваться при расширении пути. Если имя файла, которому соответствует шаблон расширения имени пути, также соответствует одному из шаблонов в GLOBIGNORE , оно удаляется из списка совпадений.

## HISTCONTROL

Переменная:	HISTCONTROL	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Разделенный двоеточием список значений, определяющих, как команды сохраняются в списке истории:

<code>ignorespace</code>	строки, начинающиеся с пробела, не сохраняются в списке истории
<code>ignoredups</code>	не сохраняйте строки, соответствующие предыдущей записи истории
<code>ignoreboth</code>	сокращение от <code>ignorespace:ignoredups</code>
<code>erasedups</code>	удалите все предыдущие строки, соответствующие текущей строке, из списка истории перед сохранением текущей строки

Любое значение, которого нет в приведенном выше списке, игнорируется.

Если `HISTCONTROL` значение не задано или не содержит допустимого значения, все строки, прочитанные анализатором оболочки, сохраняются в списке истории в зависимости от значения `HISTIGNORE` . Вторая и последующие строки многострочной составной команды не проверяются и добавляются в историю независимо от значения `HISTCONTROL` .

## HISTFILE

<b>Переменная:</b>	<code>HISTFILE</code>	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	если не задано	<b>По умолчанию:</b>	<code>~/.bash_history</code>

Имя файла, в котором сохраняется история команд.

Если значение не задано, история команд не сохраняется при выходе из интерактивной оболочки.

## HISTFILESIZE

<b>Переменная:</b>	<code>HISTFILESIZE</code>	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	если не задано	<b>По умолчанию:</b>	<code>HISTSIZE</code>

Максимальное количество строк, содержащихся в файле истории.

Когда этой переменной присваивается значение, файл истории при необходимости усекается путем удаления самых старых записей, чтобы содержать не более заданного количества строк. Если заданное количество строк равно 0 (ноль), файл обрезается до нулевого размера. Нечисловые значения и числовые значения меньше нуля препятствуют усечению.



Файл истории также обрезается до этого размера после его записи при выходе из интерактивной оболочки.

## ГИСТИНЬОРЕ

Переменная:	HISTIGNORE	Поскольку:	2.0
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Разделенный двоеточием список шаблонов, используемых для определения того, какие командные строки следует сохранять в списке истории. Каждый шаблон привязан к началу строки и должен соответствовать полной строке (неявный '\*' не добавляется).

Каждый шаблон проверяется на соответствие строке после применения проверок, указанных HISTCONTROL .

В дополнение к обычным символам соответствия шаблону оболочки, "&" соответствует предыдущей строке истории. "&" может быть экранирован с помощью обратной косой черты; обратная косая черта удаляется перед попыткой сопоставления.

Вторая и последующие строки многострочной составной команды не проверяются и добавляются в историю независимо от значения HISTIGNORE .

## HISTSIZE

Переменная:	HISTSIZE	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	если не задано	По умолчанию:	устанавливается во время компиляции (по умолчанию 500)

Количество команд, которые нужно запомнить в истории команд.

Если число установлено в 0 (ноль), то список истории отключен. Если число установлено в любое отрицательное число, то список истории не ограничен.

## Формат гистограммы

Переменная:	HISTTIMEFORMAT	Поскольку:	неизвестно
-------------	----------------	------------	------------

Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Если эта переменная установлена, а не равна null, ее значение используется в качестве строки формата для strftime(3) печати временной метки, связанной с каждой записью истории, отображаемой встроенной историей.

Если эта переменная установлена, временные метки записываются в файл истории, чтобы они могли сохраняться во время сеансов оболочки. Здесь используется символ комментария истории, чтобы отличать временные метки от других строк истории.

## ГЛАВНАЯ

Переменная:	HOME	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Домашний каталог текущего пользователя.

Аргумент по умолчанию для встроенной команды cd.

Значение этой переменной также используется при выполнении расширения тильды.

## ФАЙЛ ХОСТА

Переменная:	HOSTFILE	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Содержит имя файла в том же формате, /etc/hosts который должен быть прочитан, когда оболочке необходимо заполнить имя хоста.

Список возможных дополнений имени хоста может быть изменен во время работы оболочки. при следующей попытке завершения имени хоста после изменения значения Bash добавляет содержимое нового файла в существующий список.

Если HOSTFILE задано, но не имеет значения или не дает имени читаемому файлу, Bash пытается прочитать /etc/hosts , чтобы получить список возможных дополнений имени хоста.

Когда HOSTFILE значение не установлено, список имен хостов очищается.

## ЕСЛИ

Переменная:	IFS	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	<space><tab> <newline>

Внутренний разделитель полей, который используется для разделения слов после расширения и для разделения строк на слова с помощью встроенной команды `read` .

## ИГНОРИРОВАНИЕ

Переменная:	IGNOREEOF	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	10 (если неверно)

Управляет действием интерактивной оболочки при получении `EOF_(.)` символа (например, с помощью `Ctrl-D`) в качестве единственного ввода.

Если задано, значение представляет собой количество последовательных символов `EOF_(.)`, которые должны быть введены в качестве первых символов в строке ввода перед завершением работы `Bash`.

Если переменная существует, но не имеет числового значения или не имеет значения, значение по умолчанию равно 10.

Если он не существует, `EOF_(.)` означает конец ввода в оболочку.

## INPUTRC

Переменная:	INPUTRC	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Имя файла для файла запуска `readline`, переопределяющее значение по умолчанию `~/.inputrc` .

## LANG

Переменная:	LANG	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Используется для определения категории локали для любой категории, специально не выбранной с переменной, начинающейся с LC\_ .

## LC\_ALL

Переменная:	LC_ALL	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Эта переменная переопределяет значение LANG и любую другую LC\_ переменную, определяющую категорию локали.

## LC\_COLLATE

Переменная:	LC_COLLATE	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Эта переменная определяет порядок сортировки, используемый при сортировке результатов расширения имени пути, и определяет поведение выражений диапазона, классов эквивалентности и сопоставляющих последовательностей в рамках расширения имени пути и сопоставления с образцом.

## LC\_STYPE

Переменная:	LC_STYPE	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Эта переменная определяет интерпретацию символов и поведение классов символов в рамках расширения имени пути и сопоставления с образцом.

## LC\_MESSAGES

Переменная:	LC_MESSAGES	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Эта переменная определяет языковой стандарт, используемый для перевода строк, заключенных в двойные кавычки, перед которыми стоит а \$ .

## LC\_NUMERIC

Переменная:	LC_NUMERIC	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Эта переменная определяет категорию локали, используемую для форматирования чисел.

## СТРОКИ

Переменная:	LINES	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	на SIGWINCH	По умолчанию:	n/a

Используется составной командой select для определения длины столбца для печати списков выбора. Автоматически устанавливается при получении SIGWINCH .

## Почта

Переменная:	MAIL	Поскольку:	неизвестно
-------------	------	------------	------------

Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	зависит от системы

Если для этого параметра задано имя файла или каталога, а MAILPATH переменная не задана, Bash информирует пользователя о прибытии почты в указанный файл или директорию формата Mailedir.

## ПРОВЕРКА ПОЧТЫ

Переменная:	MAILCHECK	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	60

Указывает, как часто (в секундах) Bash проверяет почту.

Когда приходит время проверять почту, оболочка делает это перед отображением основного приглашения.

Если эта переменная не задана или установлена в значение, которое не является числом, большим или равным нулю, оболочка отключает проверку почты.

## ПОЧТОВЫЙ ПУТЬ

Переменная:	MAILPATH	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	зависит от системы

Список имен файлов, разделенных двоеточием, для проверки почты.

Сообщение, которое будет напечатано при поступлении почты в определенном файле, может быть указано путем отделения имени файла от сообщения символом "?" (вопросительный знак).

При использовании в тексте сообщения \$\_ расширяется до имени текущего почтового файла.

Пример содержимого:

```
/var/mail/bfox?"У вас есть почта": ~/shell-mail?"У $_ есть почта!"
```

# OPTERR

Переменная:	OPTERR	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	ДА	По умолчанию:	1 (устанавливается при запуске)

Если установлено значение 1, Bash отображает сообщения об ошибках, сгенерированные встроенной командой getopts.

OPTERR инициализируется значением 1 при каждом вызове оболочки или выполнении сценария оболочки.

# ПУТЬ

Переменная:	PATH	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	зависит от системы (устанавливается во время компиляции)

Путь поиска команд. Это разделенный двоеточием список каталогов, в которых оболочка ищет команды.

Имя каталога нулевой длины (null) в значении PATH указывает на текущий каталог.

Пустое имя каталога может отображаться в виде двух смежных двоеточий или в виде начального или конечного двоеточия.

Может быть статический путь, скомпилированный для использования в ограниченной оболочке.

# POSIXLY\_CORRECT

Переменная:	POSIXLY_CORRECT	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Если эта переменная находится в среде при запуске Bash, оболочка переходит в режим posix перед чтением файлов запуска, как если `-posix` бы была указана опция вызова.

Если он установлен во время работы оболочки, Bash включает режим posix, как если бы команда `set -o posix` была выполнена.

## PROMPT\_COMMAND

<b>Переменная:</b>	PROMPT_COMMAND	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Если задано, значение выполняется как команда перед выдачей каждого основного запроса.

## PROMPT\_COMMANDS

<b>Переменная:</b>	PROMPT_COMMANDS	<b>Поскольку:</b>	5.1-альфа
<b>Тип:</b>	целочисленный индексированный массив	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Если задано, каждый элемент выполняется как команда перед выдачей каждого основного приглашения (например PROMPT\_COMMAND , просто как массив).

## PROMPT\_DIRTRIM

<b>Переменная:</b>	PROMPT_DIRTRIM	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Если задано значение, большее нуля, значение используется как количество конечных компонентов каталога, которые необходимо сохранить при расширении `\w` , и `\W` строка запроса экранируется.

Удаленные символы заменяются многоточием.



## PS0

Переменная:	PS0	Поскольку:	4.4.0
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	если не задано	По умолчанию:	" "

Расширяется и отображается интерактивными оболочками после прочтения полной команды, но перед ее выполнением.

## PS1

Переменная:	PS1	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	если не задано	По умолчанию:	"\s-\v\\$"

Значение этого параметра расширяется и используется в качестве основной строки запроса. См. раздел Управление приглашением (<https://www.gnu.org/software/bash/manual/bash.html#Controlling-the-Prompt>).

## PS2

Переменная:	PS2	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	если не задано	По умолчанию:	"> "

Значение этого параметра расширяется, как и в PS1, и используется в качестве дополнительной строки запроса.

## PS3

Переменная:	PS3	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Значение этого параметра используется в качестве подсказки для команды `select` .

## PS4

<b>Переменная:</b>	PS4	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	если не задано	<b>По умолчанию:</b>	" + "

Значение этого параметра расширяется, как и в `PS1` , и значение выводится перед каждой командой, отображаемой `Bash` во время трассировки выполнения. Первый символ `PS4` реплицируется несколько раз, по мере необходимости, для обозначения нескольких уровней косвенности.

## ОБОЛОЧКА

<b>Переменная:</b>	SHELL	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	НЕТ	<b>По умолчанию:</b>	n/a

Полный путь к оболочке хранится в этой переменной окружения. Если он не установлен при запуске оболочки, `Bash` присваивает полный путь к оболочке входа текущего пользователя.

## SRANDOM

<b>Переменная:</b>	SRANDOM	<b>Поскольку:</b>	5.1-альфа
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается Bash:</b>	ДА	<b>По умолчанию:</b>	n/a

Переменная, которая выдает 32-битное случайное число. Генерация случайных чисел использует специфичные для платформы генераторы в фоновом режиме и встроенный резервный генератор.

## Формат ВРЕМЕНИ

Переменная:	TIMEFORMAT	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Значение этого параметра используется в качестве строки формата, определяющей, как должна отображаться информация о времени для конвейеров с префиксом слова `time reserved`.

Символ `%` вводит управляющую последовательность, которая расширяется до значения времени или другой информации. Управляющие последовательности и их значения следующие, фигурные скобки обозначают необязательные части:

<code>%%</code>	литерал <code>%</code> (знак процента)
<code>%[p][l]R</code>	прошедшее время в секундах
<code>%[p][l]U</code>	количество секунд процессора, потраченных в пользовательском режиме
<code>%[p][l]S</code>	количество секунд, потраченных процессором в системном режиме
<code>%P</code>	Процент процессора, вычисляемый как $(\%U + \%S) / \%R$

Необязательными модификаторами (`p` и `l`) являются:

- `p` Цифра, указывающая точность. Значение 0 не приводит к выводу десятичной точки или дроби. Отображается не более трех цифр после запятой. Если не указано, используется значение 3.
- `l` Более длинный формат, включая минуты, формы `MMmSS.FFs`. Значение `p` определяет, включена ли дробь.

Если эта переменная не задана, Bash действует так, как если бы у нее было значение

```
$'\ nreal \ t%3lR \ nuser \ t%3lU \ nsys%3lS'
```

Если значение равно `null`, информация о времени не отображается.

При отображении строки формата добавляется завершающая новая строка.

## TMOUT

Переменная:	TMOUT	Поскольку:	2.05b
Тип:	обычная переменная	Доступно только для чтения:	НЕТ

<b>Устанавливается</b>	НЕТ	<b>По умолчанию:</b>	n/a
<b>Bash:</b>			

Если установлено значение, большее нуля, `TMOUT` обрабатывается как тайм-аут по умолчанию для встроенной команды чтения.

Команда `select` завершается, если ввод не поступает через `TMOUT` несколько секунд, когда ввод поступает с терминала.

В интерактивной оболочке значение интерпретируется как количество секунд ожидания ввода после выдачи основного запроса. `Bash` завершается после ожидания в течение указанного количества секунд, если входные данные не поступают.

## TMPDIR

<b>Переменная:</b>	TMPDIR	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается</b>	НЕТ	<b>По умолчанию:</b>	n/a
<b>Bash:</b>			

Если задано, `Bash` использует его значение в качестве имени каталога, в котором `Bash` создает временные файлы для использования оболочкой.

## авто\_резум

<b>Переменная:</b>	auto_resume	<b>Поскольку:</b>	неизвестно
<b>Тип:</b>	обычная переменная	<b>Доступно только для чтения:</b>	НЕТ
<b>Устанавливается</b>	НЕТ	<b>По умолчанию:</b>	n/a
<b>Bash:</b>			

Эта переменная определяет, как оболочка взаимодействует с пользователем и управлением заданиями. Если эта переменная установлена, простые команды из одного слова без перенаправлений рассматриваются как кандидаты на возобновление существующего остановленного задания. Двусмысленность не допускается; если существует более одного задания, начинающегося с введенной строки, выбирается задание, к которому обращались последним. Имя остановленного задания в данном контексте - это командная строка, используемая для его запуска. Если задано значение `exact`, предоставленная строка должна точно соответствовать имени остановленного задания; если задано значение `substring`, предоставленная строка должна соответствовать подстроке имени остановленного задания. Значение подстроки обеспечивает функциональность, аналогичную `%?` идентификатор задания.

Если задано любое другое значение, предоставленная строка должна быть префиксом имени остановленного задания; это обеспечивает функциональность, аналогичную идентификатору %string задания.

## гистограммы

Переменная:	histchars	Поскольку:	неизвестно
Тип:	обычная переменная	Доступно только для чтения:	НЕТ
Устанавливается Bash:	НЕТ	По умолчанию:	n/a

Два или три символа, которые управляют расширением истории и токенизацией.

Первый символ - это символ расширения истории, символ, который сигнализирует о начале расширения истории, обычно "!" (восклицательный знак).

Второй символ - это символ быстрой подстановки, который используется в качестве сокращения для повторного запуска предыдущей введенной команды, заменяя одну строку другой в команде. Значение по умолчанию равно '^' (carret).

Необязательный третий символ - это символ, который указывает, что оставшаяся часть строки является комментарием, если она найдена в качестве первого символа слова, обычно '#' (хэш-метка). Символ комментария истории приводит к пропуску замены истории для оставшихся слов в строке. Это не обязательно приводит к тому, что синтаксический анализатор оболочки обрабатывает остальную часть строки как комментарий.

## Обсуждение

Чад Кломан (<http://www.cloman.com/>), 2011/04/24 04:18 (.)

У вас ошибка в написании. Это должно быть "loses" вместо "loses". Эти слова имеют два разных значения.

Ян Шампера, 2011/04/24 14:00 (.)

Спасибо, исправлено. Действительно, это странным образом меняет смысл :)

Ильгиз Латыпов, 2012/01/27 06:14 (.)

Я мог бы найти способ временно установить IFS для разделения слов в задании.

```
$ (IFS=":" m=$(echo "a: b"); echo ${m[1]}; echo -ne "${IFS}" | hexdump -C ) b
00000000 3a 7c 00000001
```

Это ограничение становится очевидным после осознания того, что присвоение `m` не является командой, поэтому Bash интерпретирует `IFS=... m= ...` как серию присвоений на текущем уровне оболочки.

Интересно, может ли Bash ввести присвоение переменных, таких как `IFS`, для разделения слов в массивах. Это будет выглядеть так,

```
m=([IFS=":"](echo "a:b"))
```

До сих пор я обходил ограничения, используя встроенную функцию чтения,

```
$ ( IFS=":" read -r m1 m2 << "a:b"; echo "${m2}"; echo -ne "$ {IFS}" | hexdump -C ) b
00000000 20 09 0a | ..| 00000003
```

swaroop vasireddy, 09.05.16 11:20 ()

Есть ли способ, которым `getopts` сможет считывать содержимое файла (т.е. Объявленных переменных) в скрипте `bash`? Если да, не могли бы вы сообщить мне об этом с примером. Это было бы большим подспорьем для меня. Очень хорошая статья и ценю ваши усилия здесь.

 syntax/shellvars.txt  Последнее редактирование: 2023/01/20 05:39 автор fgrouse

---

Этот сайт поддерживается Performing Databases - вашими экспертами по администрированию баз данных

---

Bash Hackers Wiki



Except where otherwise noted, content on this wiki is licensed under the following license:  
GNU Free Documentation License 1.3