

ФГАОУ ВО «Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики»

Кафедра вычислительной техники

Лабораторная работа №4

Системное программное обеспечение

Саржевский Иван Анатольевич

Группа Р3202

Санкт-Петербург

2019 г.

1 Первое задание

1. `grep "Sun" datebook`
2. `grep "^J" datebook`
3. `grep "700$" datebook`
4. `grep -v "834" datebook`
5. `grep ":12/" datebook`
6. `grep ":408-" datebook`
7. `grep -e "[A-Z][a-z][a-z][a-z][a-z], [A-z]" datebook`
8. `grep -e "[Kk][A-Za-z]*:" datebook`
9. `grep -e ":\$" datebook`
10. `grep "[Ll]incoln" datebook`

2 Второе задание

1. При выполнении команды `grep '\<Tom\>' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `\<Tom\>` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие слово `Tom`. Статус выполнения команды можно определить проанализировав код завершения утилиты `grep`, который сохраняется в переменной окружения `$?`
2. При выполнении команды `grep 'Tom Savage>' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `Tom Savage` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие подстроку `Tom Savage`.
3. При выполнении команды `grep '^Tommy' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `^Tommy` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, начинающиеся с `Tommy`.
4. При выполнении команды `grep '\.bak$' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `\.bak$` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, оканчивающиеся на `.bak`.
5. При выполнении команды `grep '[Pp]yramid' *` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `[Pp]yramid` и `*` соответственно. Утилита `grep` распознает аргументы и ищет во всех файлах в директории строки, содержащие подстроку `Pyramid` или `pyramid`.
6. При выполнении команды `grep '[A-Z]' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `[A-Z]` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие заглавные буквы.

7. При выполнении команды `grep '[0-9]' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `[0-9]` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие цифры.
8. При выполнении команды `grep '[A-Z]...[0-9]' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `[A-Z]...[0-9]` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие подстроку, начинающуюся с заглавной буквы, затем три любых символа и цифра.
9. При выполнении команды `grep -w '[Tt]est' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `-w`, `[A-Z]` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие слова `Test` или `test`.
10. При выполнении команды `grep -s 'Mark Todd' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `-s`, `Mark Todd` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие подстроку `Mark Todd` и подавляет сообщения о несуществующих или невалидных файлах.
11. При выполнении команды `grep -v 'Mary' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `-v`, `Mary` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, не содержащие подстроку `Mary`.
12. При выполнении команды `grep -i 'sam' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `-i`, `sam` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие подстроку `sam` без учета регистра (`Sam`, `sAm`, `SaM` etc).
13. При выполнении команды `grep -l 'Dear Boss' *` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `-l`, `Dear Boss` и `*` соответственно. Утилита `grep` распознает аргументы и выведет имена файлов, содержащих подстроку `Dear Boss`.
14. При выполнении команды `grep -n 'Tom' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `-n`, `Tom` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие заглавные буквы, затем выведет их с указанием номера строки внутри файла.
15. При выполнении команды `grep "$name" db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `$name` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие подстроку из переменной окружения `name`.
16. При выполнении команды `grep '$5' db` интерпретатор запускает утилиту `grep` передавая в качестве аргументов строки `$5` и `db` соответственно. Утилита `grep` распознает аргументы и ищет в файле `db` строки, содержащие 5 после конца строки - ничего не найдет.
17. При выполнении команды `ps -ef | grep '^ *user1'` интерпретатор запускает утилиту `grep` передавая в качестве аргумента строку `^ *user1` Утилита `grep` распознает аргументы и ищет в выводе команды `ps -ef` строки, начинающиеся с произвольного

количества пробелов, завершаемого подстрокой `user1`. Так как команда `ps -ef` вернет нам список процессов, данная команда выберет из них те, которые принадлежат пользователю `user1`.

18. При выполнении команды `egrep '^ +' db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `^ +` и `db` соответственно. Утилита `egrep` распознает аргументы и ищет в файле `db` строки, начинающиеся с произвольного количества пробелов.
19. При выполнении команды `egrep '^ *' db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `^ *` и `db` соответственно. Утилита `egrep` распознает аргументы и ищет в файле `db` строки, начинающиеся с произвольного количества пробелов (включая 0).
20. При выполнении команды `egrep '(Tom Dan) Savage' db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `(Tom Dan) Savage` и `db` соответственно. Утилита `egrep` распознает аргументы и ищет в файле `db` строки, содержащие подстроки `Tom Savage` или `Dan Savage`.
21. При выполнении команды `egrep '(ab)+'` `db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `(ab)+` и `db` соответственно. Утилита `egrep` распознает аргументы и ищет в файле `db` строки, содержащие произвольное количество повторений подстроки `ab` (`ab`, `abab` etc).
22. При выполнении команды `egrep '^X[0-9]?' db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `^X[0-9]?` и `db` соответственно. Утилита `egrep` распознает аргументы и ищет в файле `db` строки, начинающиеся с буквы `X` и одного или двух чисел.
23. При выполнении команды `egrep 'fun\.$' *` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `fun\.$` и `*` соответственно. Утилита `egrep` распознает аргументы и ищет во всех файлах директории строки, оканчивающиеся на `fun..`
24. При выполнении команды `egrep '[A-Z]+'` `db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `[A-Z]+` и `db` соответственно. Утилита `egrep` распознает аргументы и ищет в файле `db` строки, содержащие произвольное количество подряд идущих заглавных букв.
25. При выполнении команды `egrep '[0-9]'` `db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `[0-9]` и `db` соответственно. Утилита `egrep` распознает аргументы и ищет в файле `db` строки, содержащие цифры.
26. При выполнении команды `egrep '[A-Z]...[0-9]'` `db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `[A-Z]...[0-9]` и `db` соответственно. Утилита `egrep` распознает аргументы и ищет в файле `db` строки, содержащие подстроку, начинающуюся с заглавной буквы, затем три произвольных символа и цифра.
27. При выполнении команды `egrep '[tT]est'` `db` интерпретатор запускает утилиту `egrep` передавая в качестве аргументов строки `[tT]est` и `db` соответственно. Утилита

egrep распознает аргументы и ищет в файле db строки, содержащие подстроки **Test** или **test**.

28. При выполнении команды `egrep '(Suesan Jean) Doe' db` интерпретатор запускает утилиту egrep передавая в качестве аргументов строки (Suesan Jean) Doe и db соответственно. Утилита egrep распознает аргументы и ищет в файле db строки, содержащие подстроки Suesan Doe или Jean Doe.
29. `egrep -v 'Mary' db` - аналогично пункту 11.
30. `egrep -i 'sam' db` - аналогично пункту 12.
31. `egrep -l 'Dear Boss' db` - аналогично пункту 13.
32. `egrep -n 'Tom' db` - аналогично пункту 14.
33. `egrep -s "$name" db` - аналогично пункту 15, с подавлением сообщений об ошибках.