

## Блок-8. Машины (обязательные) по теме «ССЫЛОЧНЫЙ ТИП»

### Задача-1 «Выборочное копирование строк текстового файла»

Дан внешний *текстовый* файл с именем **file1.txt** (в состав файла могут входить как пустые, так и непустые строки).

За один просмотр этого файла сгенерировать другой текстовый файл с именем **file2.txt** по следующему *правилу*: во второй файл следует перенести только те строки первого файла, **длина которых больше единицы и которые при этом заканчиваются нечётной цифрой** (все остальные строки – не переносить). Файл **file1.txt** следует самостоятельно подготовить в редакторе **Jeany** (или другом редакторе, где набираете тексты программ) и прислать его мне вместе с программой.

*Требования.* В решении не использовать вспомогательных массивов, файлов. Для сохранения символов текущей строки следует воспользоваться вспомогательным *линейным списком*, в звенья которого будут последовательно заноситься (в качестве элементов) символы этой строки. При переходе к чтению следующей строки следует опустошать (с освобождением памяти) список, полученный по результату просмотра предыдущей строки.

*Рекомендации.* Описать процедуру **Create (L)**, которая строит список **L** из символов текущей строки исходного текстового файла. Описать процедуру **Move (L)**, которая переносит элементы-символы из списка **L** – в очередную строку формируемого файла. Описать также процедуру **Destroy (L)**, которая удаляет список **L** (с освобождением памяти). Остальные процедуры (если есть желание) - на усмотрение автора программы.

Файл file1.txt	Файл file2.txt
asdf	asdfg5
asdfg5	werty249
9	45
werty249	34567
45	rtYu87878787
	445
	rw9
345678	wer7891
34567	
t	
rtYu87878787	
44	
445	
2345a	
sdfgh	
rw9	
w	
wer78911	
r4	

### Задача-2 «Выборочное копирование строк текстового файла»

Дана непустая последовательность слов, в каждом из которых содержится от 1 до 6 малых латинских букв; соседние слова разделены запятой, за последним словом следует точка. Требуется ввести (с клавиатуры) эту последовательность слов в память ЭВМ, преобразовав ее во внутреннее представление (см. ниже), а затем распечатать **в алфавитном порядке**: сначала все различные однобуквенные слова с указанием для каждого из них числа его вхождений в исходную последовательность, затем аналогичным образом все различные двухбуквенные слова и т.д. В качестве внутреннего представления последовательности слов использовать: массив из 6 списков, в k-ом из которых хранятся k-буквенные слова, упорядоченные по алфавиту. Для вставки нового элемента-слова в список воспользоваться рекурсивной процедурой, написанной на семинаре (для задачи 16.24). Потребуются лишь незначительные корректировки этой процедуры – с учётом того, что в каждом звене, помимо собственно слова, нужно хранить и число его вхождений в последовательность. Для печати отдельно взятого списка (любого из шести) описать соответствующую процедуру. Ответ выдавать на экран.

**Итого: 2 обязательных задачи. Срок их сдачи – 19 декабря.**

## Блок-8. Машины (дополнительные) по теме «СПИСКИ»

Во всех задачах - не допускать утечки памяти !!!

### задача 1 (20 очков)

-----  
“Считалка” (см. 16.42 из задачника Пильщикова). Только обязательно перед началом решения прочитайте вступление к задаче 16.41 (что понимается под кольцевым списком), а также документ про другие виды списков.

Пояснение условия на примерах:

- 1) если  $n=10$ ,  $k=4$ , то последовательность удаления такая:  $e_4, e_8, e_2, e_7, e_3, e_{10}, e_9, e_1, e_6, e_5$
- 2) если  $n=10$ ,  $k=11$ , то последовательность удаления такая:  $e_1, e_3, e_6, e_{10}, e_8, e_9, e_5, e_2, e_4, e_7$
- 3) если  $n=10$ ,  $k=1$ , то последовательность удаления такая:  $e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9, e_{10}$

Решение задачи дать в соответствии со следующими требованиями:

1) Дан непустой текстовый файл (заранее самостоятельно его заготовить) с именем "friends.txt", все строки которого – непустые. В каждой его строке записано некоторое имя (совпадающих имен нет). В имени - не более 10 букв.

Описать процедуру, которая по содержимому файла строит **кольцевой однонаправленный список**, элементами которого являются прочитанные имена (недостающие буквы в имени дополнить пробелами). Каждое очередное прочитанное имя добавляется в конец формируемого списка (т.е. в первом звене - имя из 1-ой строки, в последнем звене - имя из последней строки). Последнее звено соединено ссылкой с первым.

После построения списка процедура выводит на экран количество человек в группе (по числу строк в текстовом файле), т.е. фактическое значение для  $n$ .

2) Описать процедуру, которая запрашивает у пользователя значение  $k$  (натуральное), после чего выводит на экран имена ребят (каждое - с новой строки) в порядке их удаления из круга. Процедура работает с кольцевым списком, удаляя из него на каждом шаге очередное ненужное звено (обязательно с освобождением памяти). *Внимание:* если в круге останется немного людей, а  $k$  - велико, то не следует обходить круг многократно для поиска очередного звена, подлежащего удалению.

Текстовый файл "friends.txt", над которым выполнялась Ваша программа, поместите в отправляемый архив.

### задача 2 (20 очков)

-----  
Это задача на **двунаправленный список**. Перед решением этой задачи прочтите вступление к задаче 16.43 (о том, что такое двунаправленный список и как он представляется). Условие самого задания сформулировано в номере 16.44. Требования к решению этой задачи:

- 1) Заранее (самостоятельно) подготовить текстовый файл с именем "T1.TXT", в строках которого могут появляться символы #. В файле допускаются и пустые строки.
- 2) Программа должна по файлу "T1.TXT" построить новый файл "T2.TXT". Каждая строка нового файла должна быть преобразована в соответствии с правилом, приведенным в условии упражнения 16.44.
- 3) Преобразование строки следует выполнять с использованием двунаправленного списка (сообразить самостоятельно, какие операции потребуется реализовать над этим списком).
- 4) Пустые строки или строки, ставшие пустыми в процессе преобразования, переносить в новый файл запрещено.

*Внимание:* а) в новом файле не должно быть символов #; б) в новом файле не должно быть пустых строк

5) Прислать файлы "T1.TXT" и "T2.TXT" (свой вариант). В исходном файле "T1.TXT" должно быть не менее 10 строк (среди которых - не менее двух пустых).

### **задача 3 (20 очков)**

-----

Дан непустой текстовый файл “WORDS.TXT”, в каждой строке которого записано по одному непустому слову (пробелы в записи слов не используются). Вывести на экран (в разных строках) все слова максимальной длины. Решить эту задачу за один просмотр исходного файла (просьба удерживать экран с результатами). В решении использовать **иерархический список** (в нем формируется ответ по принципу: элементы основного списка хранят ссылки на вспомогательные списки, полученные из букв слов одинаковой /максимальной длины). Не сохранять где-либо исходный текст полностью! Хранить только список из слов максимальной длины (на текущий момент обработки файла). Если выяснится, что слова из списка не подошли (появилось более длинное слово), освобождать занимаемую этими словами память и обновлять список. Поместить в отправляемый архив свой вариант файла “WORDS.TXT. Об иерархическом списке можно прочитать в присланном документе про другие виды списков.

## **И снова в бой!**