

6 зертханалық жұмыс. Стандартты математикалық функцияларды пайдалану. Тіркестермен жұмыс істеу. Типтер жүйесі. Кірістірілген типтер және оларды түрлендіру.

Тапсырмалар

1. Мысалы, `choice` айнымалысы тіркестік мәнге сілтеме жасайды делік. Төменде келтірілген `if` нұсқауы `choice` айнымалысы 'Д' немесе 'д' мәндеріне тең екендігін анықтайды:
`if choice == 'д' or choice == 'Д':`
Бұл нұсқауды тек бір салыстыру жасап, `or` операторын пайдаланатындай етіп жазыңыз. (Кеңес: `upper()` әдісін немесе `lower()` әдісін қолданыңыз).
2. `Mystring` сілтеме жасайтын тіркестік мәндегі бос таңбалардың санын анықтайтын цикл жазыңыз.
3. `Mystring` сілтеме жасайтын тіркестік мәндегі цифрлар санын есептейтін цикл жазыңыз.
4. `Mystring` сілтеме жасайтын тіркестік мәндегі төменгі регистрдегі таңбалар санын есептейтін цикл жазыңыз.
5. Аргумент ретінде тіркестік мән қабылдайтын функция жазыңыз және сол аргумент `'.com'` тіркесімен аяқталатын болса, функция ақиқат мәнін қайтаратын болсын. Қарама қарсы жағдайда, функция жалған мәнін қайтаруы керек.
6. Төменгі регистрдегі 'т' әрпінің барлық нұсқаларын жоғарғы регистрге түрлендіретін тіркестік мәнің көшірмесін жасайтын код фрагментін жазып шығыңыз.
7. Аргумент ретінде тіркестік мән қабылдап алып, соны кері бағытта жазып шығаратын функция құрыңыз.
8. Мысалы, `mystring` айнымалысы тіркестік мәнге сілтеме жасайды делік. Қиынды өрнегін қолдана отырып, тіркес мәніндегі алғашқы 3 таңбаны шығаратын нұсқау жазыңыз.
9. Мысалы, `mystring` айнымалысы тіркестік мәнге сілтеме жасайды делік. Қиынды өрнегін қолдана отырып, соңғы 3 таңбаны шығаратын нұсқау жазыңыз.
10. Төмендегі нұсқауға қараңыз:
`mystring = 'бауырсақ>сүт>құрт>ірімшік>шелпек'`
Осы тіркес мәнін төмендегі тізімге түрлендіретін нұсқау жазыңыз:
`['бауырсақ', 'сүт', 'құрт', 'ірімшік', 'шелпек']`

№8 зертханалық жұмыс. Программалауға берілген есептер

1. **Инициалдар.** Адамның аты, әкесінің аты және тегі бар тіркестік мәнді алып, содан инициал құратын программа жазыңыз. Мысалы, егер пайдаланушы Марат Исламұлы Камалов тіркесін енгізсе, онда программа М. И. К. таңбаларын шығаруы тиіс.
2. **Тіркестегі цифрлар қосындысы.** Пайдаланушыдан аралары бөлінбеген цифрлар тізбегін тіркес ретінде енгізуді сұрайтын программа жазыңыз. Программа функция арқылы тіркестегі бір таңбалы цифрлар қосындысын экранға шығаруы тиіс. Мысалы, егер пайдаланушы 2514 енгізсе, онда бұл әдіс 12 мәнін қайтару керек, өйткені ол 2, 5, 1 және 4 сандарының қосындысы.
3. **Күн-ай мерзімін (датаны) шығару.** Пайдаланушыдан `кк/аа/жжжж` форматындағы датаны (12.03.2020) бейнелейтін тіркестік мәнді алып, оны толық дата түрінде, мысалы, `2020 жылғы 12 наурыз` деп экранға шығару керек.
4. **Морзе әліппесі конвертері.** Морзе әліпбиі – әрбір алфавит әрпі, әрбір цифр және әртүрлі тыныс белгілері нүктелер және сызықшалармен берілген кодтау түрі. 8.4 және 8.5 кестелерде осы әліппе бөлігі көрсетілген. Пайдаланушыны тіркестік мән енгізуді

сұрайтын программа жазыңыз және ол осы тіркестік мәнді Морзе кодтарына сәйкес түрлендіретін болсын.

8.4 кесте. Морзе (халықаралық) әліпбиі

Символ	Код	Символ	Код	Символ	Код	Символ	Код
бос орын	<i>бос орын</i>	6	—....	G	—.	Q	—.-
үтір	—..—	7	—...—	H	R	.-.
нүкте	..-.-	8	----..	I	..	S	...—
сұрақ белгісі	..-..	9	----.	J	.----	T	-
0	-----	A	.-	K	-.-	U	..-
1	.-----	B	-...	L	.-..	V	...-
2	..----	C	-.-.	M	--	W	.---
3	...---	D	-..	N	-.	X	-.-.
4-	E	.	O	---	Y	-.-
5		F	..-.	P	.-.	Z	--..

8.5 кесте. Морзе (орыс әріптері) әліпбиі

Символ	Код	Символ	Код	Символ	Код	Символ	Код
А	.-	И	..	Р	.-.	Ш	----
Б	-...	Й	.----	С	...	Щ	--.-
В	.--	К	-.-	Т	-	Ъ	.-.-.
Г	--.	Л	.-..	У	..-	Ы	-.--
Д	-..	М	--	Ф	..-	Ь	-.-
Е, Ё	.	Н	-.	Х	Э	..-..
Ж	...-	О	---	Ц	-.-.	Ю	..--
З	--..	П	.-.	Ч	---.	Я	.-.-

5. **Телефон нөмірін алфавиттік аудармашы.** Көптеген компаниялар клиенттерге нөмірлерді есте сақтау оңай болуы үшін, 555-GET-FOOD сияқты телефон нөмірлерін пайдаланады. Стандартты телефонда алфавит әріптеріне келесідей сандар сәйкес келеді:

А, В және С = 2

Д, Е және F = 3

Г, Н және I = 4

Ж, К және L = 5

М, N және 0 = 6

Р, Q, R және S = 7

Т, U және V = 8

W, X, Y және Z = 9

Пайдаланушыдан XXX-XXX-XXXX форматында 10 символдық телефон нөмірін енгізуді сұрайтын программа жазыңыз. Программа түпнұсқадағы барлық әріптік таңбаларды солардың сандық эквиваленттеріне аударып, шыққан телефон нөмірін көрсетуі тиіс. Мысалы, егер пайдаланушы 555-GET-FOOD тіркесін енгізсе, онда программа 555-438-3663 нөмірін шығаруы керек.

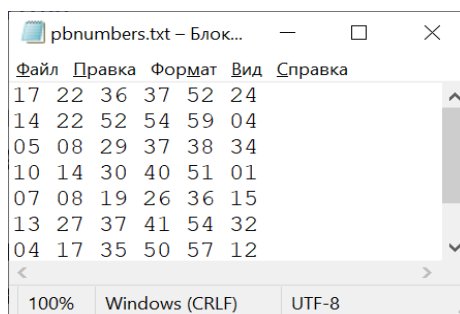
6. **Сөздердің орташа саны.** 8-тараудың бастапқы кодтарының ортасында, сондағы ішкі тараудың "Программалаудан есептер шығару" data бумасындағы `text.txt` файлы тауып алыңыз. Файлдағы әр жолда бір сөйлем сақталады. Файл мәліметін оқып, әрбір сөйлемдегі сөздердің орташа санын есептейтін программа жазыңыз.

7. **Символдарды талдау.** 8-тараудың бастапқы кодтарының ортасында, сондағы ішкі тараудың "Программалаудан есептер шығару" data бумасындағы `text.txt` файлы тауып алыңыз. Файлдағы мәліметті оқып, төмендегілерді анықтайтын программа жазыңыз:

- файлдағы жоғарғы регистрде жазылған әріптер саны;
- файлдағы төменгі регистрде жазылған әріптер саны;
- файлдағы цифрлар саны;
- файлдағы босорындар саны.

8. **Ұсыныстар корректоры.** Аргумент ретінде тіркестік мәнді қабылдап алып, ондағы әрбір сөйлемнің бірінші символын жоғарғы регистрде жазылған әріпке түрлендіріп, солардың көшірмесін қайтаратын функциясы бар программа жазыңыз. Мысалы, егер аргумент "сәлем! менің атым Азат. ал сенің атың қалай?" деген тіркес түрінде болса, онда бұл функция "Сәлем! Менің атым Азат. Ал сенің атың қалай?" деген тіркес қайтаруы керек. Программа пайдаланушыға тіркестік мәнді енгізіп, оны функцияға беру мүмкіндігін беруі тиіс. Өзгертілген тіркестік мән экранға шығарылады.
9. **Дауысты және дауыссыз әріптер.** Аргумент ретінде тіркестік мәнді қабылдап алып, сондағы дауысты әріптердің санын қайтаратын функциясы бар программа жазыңыз. Сол программада аргумент ретінде тіркестік мән қабылдап, ондағы дауыссыз әріптердің санын қайтаратын тағы бір функция болуы тиіс. Сонымен, бағдарлама пайдаланушыға тіркестік мән енгізіп, ондағы дауысты және дауыссыз әріптер санын көрсетуге мүмкіндік беруі керек.
10. **Ең жиі кездесетін символ.** Пайдаланушыға тіркестік мән енгізу мүмкіндігін беретін және сондағы ең жиі кездесетін символды экранға шығаратын программа жазыңыз.
11. **Сөздерді бөліп жазу.** Барлық сөздер арасына бос орын таңбасы қойылмай жазылған, бірақ әрбір сөздің бірінші әрпі жоғарғы регистрде (бас әріппен) терілген сөйлем қабылдайтын программа жазыңыз. Сөйлемді ондағы сөздерді босорындармен бөліп жазатын тіркеске түрлендіріңіз, онда тек сөйлемнің бірінші әрпі ғана бас әріппен жазылатын болсын. Мысалы, "СтуденттерСабақтарынаДайындалыпЖатыр" деген сөз тіркесі "Студенттер сабақтарына дайындалып жатыр" деп түрлендіріледі.
12. **Жастардың жаргоны** (жаңғалақ сөзі). Программаға бір сөйлем қабылдап алып, оның әрбір сөзін "жастардың жаргонына" түрлендіретін бағдарлама жазыңыз. Оның бір нұсқасында сөздерді жастардың жаргонына өзгерту кезінде, олардың бірінші әрпі алынып, сөздің соңына қойылады да, соған "ай" буыны қосылып жазылады. Мысал келтірейік.
Қазақ тілі: **МЕН ДҮКЕНГЕ БАРЫП КЕЛДІМ**
Жастар жаргоны: **ЕНМАЙ УКЕНГЕДАЙ АРЫПБАЙ ЕЛДІМКАЙ**
13. **PowerBall лотереясы.** Лотерея ойнау үшін 1-ден 69-ға дейінгі бес сан және 1-ден 26-ға дейінгі диапазондағы "PowerBall" саны бар билет сатып алынады. (Бұл сандарды өзіңіз таңдауыңызға немесе сіз үшін билет автоматы кездейсоқ түрде таңдауына болады). Содан кейін автомат берілген күні кездейсоқ ұтыс қатарын таңдайды. Егер алғашқы бес сан кез келген тәртіппен бірінші бес ұтыс санымен сәйкес келсе және сіздің PowerBall саныңыз ұтыс санына сәйкес келсе, онда сіз өте ірі ақша сомасын құрайтын джекпот ұтып аласыз. Егер сіздің сандарыңыз тек кейбір ұтыс сандарымен сәйкес келсе, онда сіз ұтыс нөмірлерінің қаншалықты сәйкес келгеніне байланысты аз соманы ұтасыз.

8-тараудың бастапқы кодтарының ортасында, сондағы ішкі тараудың "Программалаудан есептер шығару" data бумасындағы pbnnumbers.txt файлын тауып алыңыз, онан сіз 2010 жылғы 3 ақпан мен 2016 жылғы 11 мамыр аралығынан таңдалып алынған PowerBall ұтыс нөмірлері бар мәліметтерді табасыз (файлда 654 ұтыс сандары бар). 8.6 суретте осы файл мәліметтерінің алғашқы бірнеше жолдарының мысалы көрсетілген. Файлдағы әрбір жолда берілген күні таңдалған алты саннан тұратын жиын бар. Сандар бос орындармен бөлінген және әрбір жолдағы соңғы сан осы күнгі PowerBall саны болып табылады. Мысалы, файлдағы бірінші жол 2010 жылғы 3 ақпандағы 17, 22, 36, 37, 52-ге тең сандарды және 24-ке тең PowerBall санын көрсетеді.



8.6 сурет. pbnnumbers.txt файлы

Осы файлмен жұмыс істей отырып, төмендегі жиіліктерге байланысты сандарды:

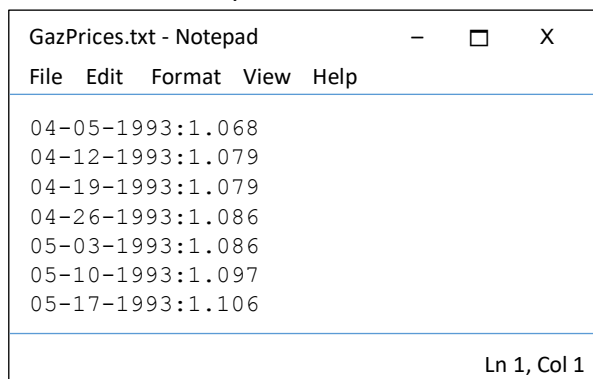
- жиілігі бойынша реттеліп келтірілген ең көп таралған 10 санды;
- жиілігі бойынша реттеліп келтірілген ең аз таралған 10 санды;
- ең "пісіп жетілген" 10 санды (ұзақ уақыт пайдаланылмаған сандар), олар да жиілігі бойынша көбінен азына қарай реттеліп орналасқан болуы тиіс;
- 1-ден 69-ға дейінгі әрбір санның кездесу жиілігін және 1-ден 26-ға дейінгі әрбір PowerBall санының жиілігін көрсететін бір немесе бірнеше бағдарлама жазып шығу керек.

14. Бензин бағалары. 8-тараудың бастапқы кодтарының ортасында, сондағы ішкі тараудың "Программалаудан есептер шығару" data бумасындағы GasPrices.txt файлын тауып алыңыз. Бұл файл 1993 жылғы 5 сәуірден бастап 2013 жылғы 26 тамызға дейінгі АҚШ-тағы бензин галлоны үшін апталық орташа бағаны көрсетеді. 8.7 суретте мысал ретінде осы файлдың алғашқы бірнеше жолдары келтірілген.

Файлдағы әрбір жол көрсетілген күнгі бензин галлоны үшін орташа бағаны қамтиды және оның форматы былайша көрсетілген:

AA-КК-ЖЖЖЖ: Баға

мұндағы **AA** – екі таңбалы ай; **КК** – екі таңбалы күн; **ЖЖЖЖ** – төрт таңбалы жыл; **Баға** – бұл көрсетілген күнгі бензин галлонының орташа бағасы.



8.7 сурет. GasPrices.txt файлының мәліметтері

Осы тапсырма аясында осы файлдың мәліметтерін оқи отырып, төменде келтірілген есептеулерді орындайтын бір немесе бірнеше бағдарлама жазу қажет.

- **Бір жыл ішіндегі орташа баға:** әрбір жыл ішіндегі бензиннің орташа бағасын есептейді. (Файл деректері 1993 жылдың сәуір айында басталып, 2013 жылдың тамыз айында аяқталады. 1993-2013 жылдар аралығында берілген мәліметтерді пайдаланыңыз.)
 - **Бір айдағы орташа баға:** файлдағы әрбір айдың орташа бағасын есептейді.
 - **Бір жылдағы ең жоғарғы және ең төменгі баға:** файлдағы әр жыл ішінде ең төменгі және ең жоғарғы бағаның күні мен шамасын анықтайды.
 - **Өсуі бойынша реттелген бағалар тізімі:** күндері мен бағалары өсу тәртібімен сұрыпталып орналастырылған мәтіндік файл жасайды.
 - **Кемуі бойынша реттелген бағалар тізімі:** күндері мен бағалары кему тәртібімен сұрыпталып орналастырылған мәтіндік файл жасайды.
- Осы есептеулердің барлығын орындау үшін бір бағдарлама немесе әрбір есептеуге арналған әртүрлі бағдарламалар жазуға болады.

Алгоритмический тренажер

1. Допустим, что переменная `choice` ссылается на строковое значение. Приведенная ниже инструкция `if` определяет, равна ли переменная `choice` значениям `'Д'` или `'д'`:
`if choice == 'Д' or choice == 'д':`
 Перепишите эту инструкцию так, чтобы она делала всего одно сравнение и использовала оператор `or`. (Подсказка: примените метод `upper()` либо метод `lower()`.)
2. Напишите цикл, который подсчитывает количество пробельных символов в строковом значении, на которое ссылается `mystring`.
3. Напишите цикл, который подсчитывает количество цифр в строковом значении, на которое ссылается `mystring`.
4. Напишите цикл, который подсчитывает количество символов в нижнем регистре в строковом значении, на которое ссылается `mystring`.
5. Напишите функцию, которая принимает строковое значение в качестве аргумента и возвращает истину, если аргумент заканчивается подстрокой `' . сот'`. В противном случае функция должна вернуть ложь.
6. Напишите фрагмент кода, делающий копию строкового значения, в котором все вхождения буквы `'т'` в нижнем регистре преобразованы в верхний регистр.
7. Напишите функцию, которая принимает строковое значение в качестве аргумента и показывает строковое значение в обратном порядке.
8. Допустим, что переменная `mystring` ссылается на строковое значение. Напишите инструкцию, которая применяет выражение среза и показывает первые 3 символа в строковом значении.
9. Допустим, что переменная `mystring` ссылается на строковое значение. Напишите инструкцию, которая применяет выражение среза и показывает последние 3 символа в строковом значении.
10. Взгляните на приведенную ниже инструкцию:
`mystring = 'пирожки>молоко>стряпня>яблочный пирог>мороженое'`
 Напишите инструкцию, которая разбивает это строковое значение, создавая приведенный ниже список:
`['пирожки', 'молоко', 'стряпня', 'яблочный: пирог', 'мороженое']`

Задачи для программирования

1. Инициалы. Напишите программу, которая получает строковое значение, содержащее имя, отчество и фамилию человека и показывает инициалы. Например, если пользователь вводит Михаил Иванович Кузнецов, то программа должна вывести М.И.К.

- Сумма цифр в строке. Напишите программу, которая просит пользователя ввести ряд однозначных чисел без разделителей. Программа должна вывести на экран сумму всех однозначных чисел в строковом значении. Например, если пользователь вводит 2514, то этот метод должен вернуть значение 12, которое является суммой 2, 5, 1 и 4.
- Принтер дат. Напишите программу, которая считывает от пользователя строковое значение, содержащее дату в формате дд/мм/гггг. Она должна напечатать дату в формате 12 марта 2018 г.
- Конвертер азбуки Морзе.** Азбука Морзе представляет собой кодировку, где каждая буква алфавита, каждая цифра и различные знаки препинания представлены серией точек и тире. В табл. 8.4 и 8.5 показана часть этой азбуки.

Напишите программу, которая просит пользователя ввести строковое значение и затем преобразует это строковое значение в кодировку азбукой Морзе.

Таблица 8.4. Азбука Морзе (интернациональная)

Символ	Код	Символ	Код	Символ	Код	Символ	Код
пробел	<i>пробел</i>	6	-....	G	-.	Q	-.-
запятая	-..-	7	--...	H	R	.-.
точка	.-.-.-	8	---..	I	..	S	...
знак вопроса	..-..	9	----.	J	.-...	T	-
0	-----	A	.-	K	-.-	U	..-
1	.-----	B	-...	L	.-..	V	...-
2	..----	C	-.-	M	--	W	.-..
3	...--	D	-..	N	-.	X	-.-.
4-	E	.	O	---	Y	-.-
5		F	..-	P	.-.	Z	--..

Таблица 8.5. Азбука Морзе (русские буквы)

Символ	Код	Символ	Код	Символ	Код	Символ	Код
А	.-	И	..	Р	.-.	Ш	----
Б	-...	Й	.-...	С	...	Щ	-.-.
В	.-..	К	-.-	Т	-	Ъ	.-.-.
Г	--.	Л	.-..	У	..-	Ы	.-..
Д	-..	М	--	Ф	..-	Ь	-.-.
Е, Ё	.	Н	-.	Х	Э	..-..
Ж	...-	О	---	Ц	-.-	Ю	..--
З	--..	П	.-.	Ч	---.	Я	.-.-

- Алфавитный переводчик номера телефона.** Многие компании используют телефонные номера наподобие 555-GET-FOOD, чтобы клиентам было легче запоминать эти номера. На стандартном телефоне буквам алфавита поставлены в соответствие числа следующим образом:

А, В и С = 2

Д, Е и F = 3

Г, Н и I = 4

Ж, К и L = 5

M, N и O = 6

P, Q, R и S = 7

T, U и V = 8

W, X, Y и Z = 9

Напишите программу, которая просит пользователя ввести 10-символьный номер телефона в формате XXX-XXX-XXXX. Приложение должно показать номер телефона, в котором все буквенные символы в оригинале переведены в их числовой эквивалент. Например, если пользователь вводит 555-GET-FOOD, то приложение должно вывести 555-438-3663. 6. Среднее количество слов. Среди исходного кода главы 8, а также в подпапке data "Решений задач по программированию" соответствующей главы вы найдете файл text.txt. В нем в каждой строке хранится одно предложение. Напишите программу, которая читает содержимое файла и вычисляет среднее количество слов в расчете на предложение.

7. **Анализ символов.** Среди исходного кода главы 8, а также в подпапке data "Решений задач по программированию" соответствующей главы вы найдете файл text.txt. Напишите программу, которая читает содержимое файла и определяет:
 - количество букв в файле в верхнем регистре;
 - количество букв в файле в нижнем регистре;
 - количество цифр в файле;
 - количество пробельных символов в файле.
8. **Корректор предложений.** Напишите программу с функцией, принимающей в качестве аргумента строковое значение и возвращающей его копию, в котором первый символ каждого предложения написан в верхнем регистре. Например, если аргументом является "привет! меня зовут джо. а как твоё имя?", то эта функция должна вернуть строковое значение 'Привет! Меня зовут Джо. А как твоё имя?'. Программа должна предоставить пользователю возможность ввести строковое значение и затем передать его в функцию. Модифицированное строковое значение должно быть выведено на экран.
9. **Гласные и согласные.** Напишите программу с функцией, которая в качестве аргумента принимает строковое значение и возвращает количество содержащихся в нем гласных. Приложение должно иметь еще одну функцию, которая в качестве аргумента принимает строковое значение и возвращает количество содержащихся в нем согласных. Приложение должно предоставить пользователю возможность ввести строковое значение и показать содержащееся в нем количество гласных и согласных.
10. **Самый частотный символ.** Напишите программу, которая предоставляет пользователю возможность ввести строковое значение и выводит на экран символ, который появляется в нем наиболее часто.
11. **Разделитель слов.** Напишите программу, которая на входе принимает предложение, в котором все слова написаны без пробелов, но первая буква каждого слова находится в верхнем регистре. Преобразуйте предложение в строковое значение, в котором слова отделены пробелами, и только первое слово начинается с буквы в верхнем регистре.

Например, строковое значение "ОстановисьИПочувствуйЗапахРоз" будет преобразовано в "Остановись и почувствуй запах роз".

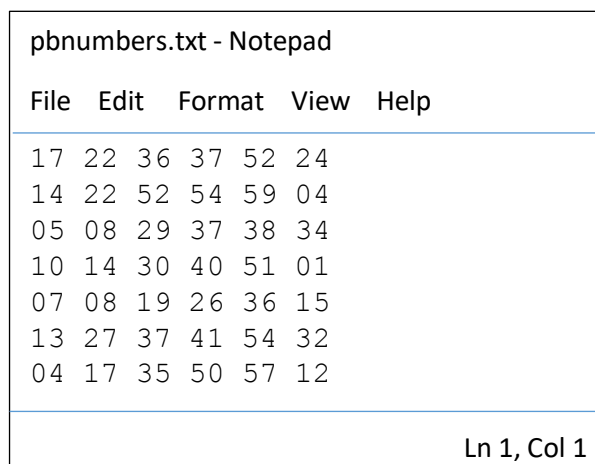
12. **Молодежный жаргон.** Напишите программу, которая на входе принимает предложение и преобразует каждое его слово в "молодежный жаргон". В одной из его версий во время преобразования слова в молодежный жаргон первая буква удаляется и ставится в конец слова. Затем в конец слова добавляется слог "ки". Вот пример.

Русский язык: ПРОСПАЛ ПОЧТИ ВСЮ НОЧЬ

Молодежный жаргон: РОСПАЛПКИ ОЧТИПКИ СЮВКИ ОЧЬНКИ

13. **Лотерея PowerBall.** Для того чтобы сыграть в лотерею PowerBall, покупают билет, в котором имеется пять чисел от 1 до 69 и число "PowerBall" в диапазоне от 1 до 26. (Эти числа можно выбрать самому либо дать билетному автомату их выбрать за вас случайным образом.) Затем в заданный день автомат случайным образом отбирает выигрышный ряд чисел. Если первые пять чисел совпадают с первыми пятью выигрышными числами в любом порядке, и ваше число PowerBall соответствует выигрышному числу PowerBall, то вы выигрываете джекпот, который составляет очень крупную сумму денег. Если ваши числа совпадают лишь с некоторыми выигрышными числами, то вы выигрываете меньшую сумму в зависимости от того, сколько выигрышных номеров совпало.

Среди исходного кода главы 8, а также в подпапке data "Решений задач по программированию" соответствующей главы вы найдете файл с именем pbnumbers.txt, содержащий выигрышные номера PowerBall, которые были отобраны между 3 февраля 2010 и 11 мая 2016 (файл содержит 654 наборов выигрышных чисел). Рис. 8.6 показывает пример первых нескольких строк содержимого этого файла. Каждая строка в файле содержит набор из шести чисел, которые были выбраны в заданную дату. Числа отделены пробелом, и последнее число в каждой строке является числом PowerBall для этого дня. Например, первая строка в файле показывает числа за 3 февраля 2010, которые равнялись 17, 22, 36, 37, 52, и число PowerBall, равное 24.



```
pbnumbers.txt - Notepad
File Edit Format View Help
17 22 36 37 52 24
14 22 52 54 59 04
05 08 29 37 38 34
10 14 30 40 51 01
07 08 19 26 36 15
13 27 37 41 54 32
04 17 35 50 57 12
Ln 1, Col 1
```

РИС. 8.6. Файл pbnumbers.txt

Напишите одну или несколько программ, которые работают с этим файлом и показывают:

- 10 наиболее распространенных чисел, упорядоченных по частоте;
- 10 наименее распространенных чисел, упорядоченных по частоте;

- 10 наиболее "созревших" чисел (чисел, которые не использовались долгое время), упорядоченных от наиболее созревших до наименее созревших;
- частоту каждого числа от 1 до 69 и частоту каждого PowerBall числа от 1 до 26.

14. **Цены на бензин.** Среди исходного кода главы 8, а также в подпапке data "Решений задач по программированию" соответствующей главы, вы найдете файл GasPrices.txt. Этот файл содержит еженедельные средние цены за галлон бензина в США, начиная 5 апреля 1993 года и заканчивая 26 августа 2013 года. На рис. 8.7 показан пример первых нескольких строк данного файла.

Каждая строка в файле содержит среднюю цену за галлон бензина в указанный день и отформатирована следующим образом:

ММ-ДД-ГГГГ :Цена

где ММ - двухзначный месяц; ДД - двухзначный день; ГГГГ - четырехзначный год; Цена - это средняя цена галлона бензина в указанный день.

GazPrices.txt - Notepad				
File	Edit	Format	View	Help
04-05-1993:1.068				
04-12-1993:1.079				
04-19-1993:1.079				
04-26-1993:1.086				
05-03-1993:1.086				
05-10-1993:1.097				
05-17-1993:1.106				
Ln 1, Col 1				

РИС. 8.7. Файл GasPrices.txt

В рамках этого задания необходимо написать одну или несколько программ, которые считывают содержимое данного файла и выполняют приведенные ниже вычисления.

- **Средняя цена за год:** вычисляет среднюю цену бензина за год для каждого года в файле. (Данные файла начинаются апрелем 1993 и заканчиваются августом 2013. Используйте данные, предоставленные за период с 1993 по 2013 годы.)
- **Средняя цена за месяц:** вычисляет среднюю цену в каждом месяце в файле.
- **Наибольшая и наименьшая цены в году:** в течение каждого года в файле определяет дату и величину самой низкой и самой высокой цены.
- **Список цен, упорядоченный по возрастанию:** генерирует текстовый файл, в котором даты и цены отсортированы в возрастающем порядке.
- **Список цен, упорядоченный по уменьшению:** генерирует текстовый файл, в котором даты и цены отсортированы в убывающем порядке.

Для выполнения всех этих вычислений можно написать одну программу или несколько разных программ, одну для каждого вычисления.

1-қосымша. Тіркестерді өңдеу әдістері

find – тіркесі ішіндегі сөздің (сөйлемнің) индексін табады – алғашқы символ индексін немесе (табылмаса) -1 мәнін қайтарады:

```
>>> s = 'The find method finds a substring'
>>> s.find('find')
4
>>> s.find('finds')
16
>>> s.find('findsa')
-1
```

join — тіркестер жиымын көрсетілген бөлгіш таңба арқылы біріктіреді:

```
>>> seq = ['one','two','three']
>>> sep = ','
>>> sep.join(seq)
'one,two,three'
```

split – бұл join әдісіне кері функция, тіркесті сөздерге (тізбектерге) бөліп береді:

```
>>> s = '/usr/local/bin'
>>> s.split('/')
['', 'usr', 'local', 'bin']
```

replace – негізгі тіркес ішіндегі бір ішкі тіркесті басқа бір тіркеске алмастырады:

```
>>> s = 'replace method returns a string'
>>> s.replace('returns','return')
'replace method return a string'
```

strip – тіркестің сол жағындағы және оң жағындағы бос орындарды өшіреді:

```
>>> '  this is whitespace string '.strip()
'this is whitespace string'
```

translate – replace әдісіне қарағанда, көптеген алмастыруларды бірден орындай алады. Келесі мысалда берілген тіркестегі әрбір '1' символы '3' символына, ал '2' символы – '4' символына алмастырылады:

```
>>> from string import maketrans
>>> table = maketrans('12', '34')
>>> '1212 5656'.translate(table)
3434 5656
```

str, int, ord, chr – әртүрлі типтерді тіркестік типке түрлендіру функциялары:

- **str** – санды тіркеске түрлендіреді;
- **int** – тіркесті санға түрлендіреді;
- **ord** – символдың сандық мәнін (кодын) қайтарады;
- **chr** – санды (кодты) символға түрлендіреді.

Python 3.0 тіліндегі тіркестік әдістер

```
S.capitalize() # Бірінші әріпті бас әріп етіп жазады
S='программалау'
print(S.capitalize()) # Программалау
S.lower() # Барлық тіркесті кіші әріппен жазып шығады
S='ПРОГРАММАЛАУ'
print(S.lower()) # программалау

S.rstrip() # тіркестің соңындағы бос орындарды өшіреді
S.strip() # тіркестің 2 жағындағы бос орындарды өшіреді
S.lstrip() # тіркестің алдындағы бос орындарды өшіреді
S=' Программалау'
print(S.lstrip()) # Программалау
find(S, start, end) # str ішкі тіркесін негізгі тіркестің start индексінен end
# индексіне дейін іздейді. Егер str табылмаса, -1 мәнін береді

S = "Hello world! Goodbye world!"
index = S.find("wor", 0, len(S))
print(index) # 6 индексін береді
```

Python 3.0 тіліндегі тіркестік әдістер

```
S.replace(ескі, жаңа) # Мұнда барлық ескі тіркестер жаңа тіркестермен
# алмастылып, шыққан тіркестің көшірмесі алынады
S.isalnum() # Тіркес цифрлар мен әріптерден тұрса, нәтижесі
# ақиқат болады
S.isalpha() # Тіркес әріптерден тұрса, нәтижесі ақиқат болады
S.isdigit() # Тіркес цифрлардан тұрса, нәтижесі ақиқат болады
S.upper() # Тіркесті жоғарғы регистрге түрлендіру
S.lower() # Тіркесті төменгі регистрге түрлендіру
S.swapcase() # Төменгі регистрдегі символдарды жоғарғы регистрге, ал
# төменгі регистрдегі символдарды ге түрлендіру
S.islower() # Тіркес төменгі регистр символдарынан тұрса, нәтижесі
# ақиқат болады
len(S) # Тіркес ұзындығын анықтайды
S[i:j:step] # Индекстері i және j аралығындағы тіркес қиындысын алады
S1 + S2 # Тіркестерді біріктіру (конкатенация)
S.center(width) # Ортаға жылжытылған тіркесті қайтарады, шеттері
# бос орынмен толтырылады
```

```
S.count(sub,start,end) # sub ішкі тіркесінің негізгі S тіркесінің start индексінен end
# индексіне дейін неше рет кездесетінін анықтайды. start
# және end міндетті емес қиынды ретіндегі аргументтер.
# start және end жазылмаса, тіркесті толық тексереді
# Мысал:
```

```
str = "бұл тіркес мысалы....!!!"
sub = 'c'
print ("str.count('c') : ", str.count(sub))
sub = 'кеc'
print ("str.count('кеc', 3, len(str)) : ", str.count(sub, 3,
```

Бұл программа нәтижесі:

```
str.count('c') : 2
str.count('кеc', 3, len(str)) : 1
```