

## Лабораторная работа № 1. Управление процессами в ОС GNU Linux

### Приложение А. Общая постановка задачи.

Требуется: разработать программу в виде Linux-приложения, которая представляет собой родительский процесс. Результат выполнения выводится на терминал/консоль. Программа должна быть устойчива к некорректному пользовательскому вводу. В следующих вариантах заданий оговаривается только функционал программы, представляющей собой дочерний процесс.

**Вариант 1.** Программа принимает от пользователя два беззнаковых целых числа  $L$  и  $U$ , каждое из которых не может состоять из более, чем 6 знаков.  $L$  – это нижняя граница диапазона,  $U$  – верхняя граница диапазона. Затем она осуществляет поиск «счастливых билетов»<sup>1</sup> из этого диапазона, когда а) сумма цифр на 1-3 позиции равна сумме цифр, стоящих на 4-6 позиции; б) сумма цифр на нечетных позициях равна сумме цифр на четных позициях; в) арифметический квадратный корень из суммы шести цифр номера билета есть целое число. На экран должны выводиться полученные значения, а также количество найденных «счастливых чисел».

**Вариант 2.** Программа принимает от пользователя номер  $k$  обобщенного числа Фибоначчи (ОЧФ)<sup>2</sup> и его порядок  $p$ , причем  $p, k > 0$ , а затем вычисляет значение ОЧФ с заданными номером  $k$  и порядком числа  $p$ , по формуле

$$F_k^p = \begin{cases} \sum_{i=k-1-p}^{k-1} F_i^p, & \text{если } k > p+1 \\ 1, & \text{если } 1 \leq k \leq p+1 \end{cases}$$

и выводит его на экран.

Таким образом шестое ОЧФ первого порядка равно 8, а восьмое ОЧФ третьего порядка равно 25.

**Вариант 3.** Программа принимает от пользователя строку, хранящую целое число со знаком, и выводит на экран эквивалент этого числа прописью<sup>3</sup>. Например, ввод «-1211» должен приводить к выводу «минус одна тысяча двести одиннадцать», а ввод «100000011» приводит к выводу «сто миллионов одиннадцать». Ограничения накладываются только представлениями максимально и минимально возможного знакового целого числа используемого языка программирования.

**Вариант 4.** Первая программа принимает от пользователя строку, хранящую вещественное число в обычном или экспоненциальном форматах<sup>4</sup>, и выводит на

1 [http://ru.wikipedia.org/wiki/Счастливый\\_билет](http://ru.wikipedia.org/wiki/Счастливый_билет)

2 В литературе также встречается термин  $p$ -число Фибоначчи. Относительно чисел первого порядка ( $p = 1$ ) см. [http://ru.wikipedia.org/wiki/Числа\\_Фибоначчи](http://ru.wikipedia.org/wiki/Числа_Фибоначчи)

3 Пример реализации - <http://algotlist.manual.ru/math/misc/sumwrite.php>

4 Подробнее о форматах плавающих чисел - [http://ru.wikipedia.org/wiki/Плавающая\\_запятая](http://ru.wikipedia.org/wiki/Плавающая_запятая)

экран строковый эквивалент этого числа прописью (например, ввод «-12.11» должен приводить к выводу «минус двенадцать целых одиннадцать сотых», а ввод «1,2E+2» приводит к выводу «120 целых»). Ограничения: 1) для обычного формата после «точки» не более двух знаков; 2) для экспоненциального формата после «точки» не более шести знаков; 3) до точки не более 15 знаков для обычного формата; 4) знак «точки» - это «.»(точка) или «,» (запятая).

**Вариант 5.** Программа принимает от пользователя две строки. Далее, если обе строки хранят целые числа со знаком, то на экран выводится сумма чисел. Если были переданы вещественные числа в обычном или экспоненциальном форматах, то на экран выводится остаток от деления первого числа на второе. Во всех остальных случаях выводится результат конкатенации двух полученных строк.

**Вариант 6.** Программа принимает от пользователя два целых числа  $A$  и  $B$ , где  $A$  — сдвигаемое число,  $B$  — величина логического, арифметического или циклического сдвига<sup>1</sup>. После этого осуществляется логический, арифметический и циклический сдвиг числа  $A$  на величину  $B$ , и три результата выводятся на экран.

**Вариант 7.** Программа принимает от пользователя некое слово, после чего проверяет его на корректность (отсутствие знаков препинания, цифр). Затем программа проверяет, является ли введенное слово палиндромом<sup>2</sup> и информирует пользователя о результате проверки. Учесть смешение и «похожесть» латинских и кириллических символов. Например, следует считать латинскую  $A$  эквивалентной кириллическому символу  $A$ .

**Вариант 8.** Программа принимает от пользователя две даты — строки вида ЦЦ.ЦЦ.ЦЦЦЦ, где Ц — это любая десятичная цифра. Далее она вычисляет полное количество дней, прошедших между двумя введенными датами, и выводит его на экран. Программа должна принимать во внимание некорректные даты, например, 37.06.2006 или 01.18.2010. Следует также учесть такие особенности Григорианского календаря<sup>3</sup>, как переход на него в России в 1918 году (отсутствие дней с 1 по 13 февраля). В качестве базовой даты принять 1 января 1601.

**Вариант 9.** Программа принимает от пользователя два значения времени суток — строки вида ЦЦ.ЦЦ.ЦЦ, где Ц — это любая десятичная цифра. Далее программа вычисляет полное количество секунд, прошедших между двумя значениями времени, и выводит его на экран. Программа должна принимать во

1 Битовый сдвиг - [http://ru.wikipedia.org/wiki/Битовый\\_сдвиг](http://ru.wikipedia.org/wiki/Битовый_сдвиг)

2 Палиндром - <http://ru.wikipedia.org/wiki/Палиндром>

3 Григорианский календарь - [http://ru.wikipedia.org/wiki/Григорианский\\_календарь](http://ru.wikipedia.org/wiki/Григорианский_календарь)

внимание некорректные значения, например, 28.00.06, или 01.99.20, или 08.08.65.

**Вариант 10.** Программа принимает от пользователя две строки, осуществляет поиск количества вхождений второй строки в первую любым известным методом, кроме прямого (линейного, грубой силы), и выводит на экран полученное значение<sup>1</sup>.

**Вариант 11.** Программа принимает от пользователя дату – строку вида ЦЦ.ЦЦ.ЦЦЦЦ, где Ц – это любая десятичная цифра, и выводит на экран число, месяц и год прописью, а затем – слово «года» (например, ввод «29.02.2012» приводит к выводу «Двадцать девятое февраля две тысячи двенадцатого года»). Программа должна принимать во внимание некорректные даты, например, 37.06.2006 или 01.18.2006. Следует также учесть такие особенности Григорианского календаря<sup>2</sup>, как переход на него в России в 1918 году (отсутствие дней с 1 по 13 февраля). В качестве базовой даты принять 1 января 1601.

**Вариант 12.** Программа принимает от пользователя значение времени суток – строку вида ЦЦ.ЦЦ.ЦЦ,Ц или ЦЦ.ЦЦ.ЦЦ,ЦЦ, где Ц – это любая десятичная цифра, и выводит на экран значение часов минут, секунд и их долей прописью (например, ввод «12.01.20,15» приводит к выводу «двенадцать часов одна минута двадцать и пятнадцать сотых секунды»). Программа должна принимать во внимание некорректные значения, например, 28.00.06,14; 01.99.20,15; 08.08.65,16. При реализации достаточно обеспечить обработку десятых или сотых долей секунды, более точные значения — по инициативе разработчика.

**Вариант 13.** Программа принимает от пользователя беззнаковое целое десятичное число  $N$  – основание системы счисления ( $N > 1$  и  $N \leq 20$ ) и последовательность цифр в соответствии с заданной системой счисления. Затем программа выводит число на экран, переводит его в десятичную систему, выводит на экран, дополняет его (инвертирует) до максимальной цифры в заданной системе счисления, выводит на экран значение инвертированной последовательности, переводит ее в десятичную систему и выводит полученное число на экран.

**Вариант 14.** Программа принимает от пользователя беззнаковое целое десятичное число  $N$  – основание системы счисления ( $N > 1$  и  $N \leq 20$ ) и последовательность цифр в соответствии с заданной системой счисления. Затем программа выводит число на экран, переводит его в десятичную систему, выводит на экран, осуществляет его реверс (меняет порядок следования знаков

<sup>1</sup> Точный поиск подстроки в строке - <http://algolist.manual.ru/search/esearch/>

<sup>2</sup> Григорианский календарь - [http://ru.wikipedia.org/wiki/Григорианский\\_календарь](http://ru.wikipedia.org/wiki/Григорианский_календарь)

на обратный), выводит на экран значение измененной последовательности, переводит ее в десятичную систему и выводит его на экран.

**Вариант 15.** Программа принимает от пользователя три строки, (первая и третья строки – это правильные рациональные или десятичные дроби вида «1/3» или «0,5», вторая строка – это знак арифметической операции вида «+», «-», «\*», «/» либо операции сравнения «<», «>», «=», «!=», «>=», «<=»), выполняет требуемую операцию над полученными операндами, и выводит результат на экран. Обеспечить также сокращение дроби при необходимости. Если оба операнда арифметической операции являются рациональными дробями, результатом тоже должна быть рациональная дробь. Для операций сравнения достаточно результата «Истина» или «Ложь».

**Вариант 16.** Программа принимает от пользователя две квадратные матрицы одинакового размера, значение которого также вводит пользователь. а затем выводит на экран сумму матриц<sup>1</sup> и значение ее определителя<sup>2</sup>.

**Вариант 17.** Программа принимает от пользователя квадратную матрицу<sup>1</sup>, а также два числа – номер столбца и номер строки, затем «вычеркивает» из матрицы столбец с номером  $N$  и строку с номером  $M$ , выводит измененную матрицу на экран, вычисляет и выводит на экран значение ее определителя<sup>2</sup>.

**Вариант 18.** Программа принимает от пользователя элементы целочисленного одномерного массива, а также целочисленное значение для поиска, затем осуществляет поиск элемента двумя любыми известными методами, кроме прямого (линейного) поиска<sup>3</sup>, и выводит результат на экран.

**Вариант 19.** Программа принимает от пользователя последовательность символьных строк, осуществляет смену регистра всех букв английского алфавита, а также замену символов табуляции двумя пробелами, удаляет хвостовые пробелы, выполняя эти операции везде, за исключением подстрок, заключенных в кавычки или апострофы, и выводит результат на экран.

**Вариант 20.** Программа принимает от пользователя коэффициенты квадратного уравнения ( $ax^2+bx+c=0$ ) или кубического уравнения ( $ax^3+bx^2+cx+d=0$ ), каждый из которых не равен нулю или единице. Затем программа выводит на экран разложение кубического или квадратного уравнения на множители<sup>4</sup>.

---

1 Матрица (математика) - [http://ru.wikipedia.org/wiki/Матрица\\_\(математика\)](http://ru.wikipedia.org/wiki/Матрица_(математика))

2 Определитель - [http://ru.wikipedia.org/wiki/Определитель\\_матрицы](http://ru.wikipedia.org/wiki/Определитель_матрицы)

3 Алгоритмы поиска - [http://ru.wikipedia.org/wiki/Категория:Алгоритмы\\_поиска](http://ru.wikipedia.org/wiki/Категория:Алгоритмы_поиска)

4 Формулы Виета - [http://ru.wikipedia.org/wiki/Формулы\\_Виета](http://ru.wikipedia.org/wiki/Формулы_Виета)

**Вариант 21.** Программа принимает от пользователя две или три строки. При вводе трех строк первая и третья – это комплексные числа<sup>1</sup> вида « $\pm 1.2.\pm i5.4$ » (моделируется программно двумя вещественными и/или целыми числами), причем первый знак необязателен, вторая строка – это знак операции вида «+», «-», «\*», «/». При вводе двух строк первая — комплексное число, вторая — операция, обозначенная литерами  $S$  (комплексно-сопряженное число),  $R$  (вещественная часть),  $I$  (мнимая часть). Затем программа выполняет требуемую операцию с полученными операндами, и выводит результат на экран.

**Вариант 22.** Программа принимает от пользователя целочисленную квадратную матрицу<sup>2</sup>, после этого заменяет строку с минимальным элементом нулями, а затем **удаляет** путем перемещения элементов столбец с максимальным элементом. На экран выводится результирующая матрица, номера модифицированной строки и удаленного столбца.

**Вариант 23.** Программа принимает от пользователя целочисленную матрицу<sup>2</sup>. Упорядочить строки матрицы в порядке убывания суммы модулей их элементов. На экран выводится результирующая матрица и значения сумм модулей элементов каждой строки.

**Вариант 24.** Программа принимает от пользователя строку, содержащую дату в формате ДД.ММ.ГГГГ и выводящую на экран день недели, соответствующий введенной дате (для 03.10.2011 — «понедельник»)<sup>3</sup>. Необходимо учесть ввод некорректных значений (например, 36.13.2006 или 04.02.1918<sup>4</sup>). В качестве базовой даты принять 1 января 1601.

**Вариант 25.** Программа принимает от пользователя три строки: 1) дата в формате ДД.ММ.ГГ; 2) местное время (Красноярск, UTC +8) в формате ЧЧ.ММ; 3) Часовой пояс  $N$ , в котором необходимо узнать время и дату в формате, где  $N$  – число от -12 до +14. Программа должна вывести на экран дату и время в выбранном часовом поясе. Необходимо учесть переход в следующие сутки (например, когда в Красноярске 01.03.2010, 08.15 в Мехико 28.02.2010, 19.15), а также ввод некорректных значений (например, 36.13.2006, 25.15). В качестве базовой даты принять 1 января 1901.

**Вариант 26.** Программа принимает от пользователя две строки и выводит минимальное количество стираний, замен и добавлений символов, преобразующих одну строку в другую<sup>5</sup>.

1 Комплексное число - [http://ru.wikipedia.org/wiki/Комплексное\\_число](http://ru.wikipedia.org/wiki/Комплексное_число)

2 Матрица (математика) - [http://ru.wikipedia.org/wiki/Матрица\\_\(математика\)](http://ru.wikipedia.org/wiki/Матрица_(математика))

3 Алгоритм вычисления дня недели - [http://ru.wikibooks.org/wiki/Алгоритм\\_вычисления\\_дня\\_недели](http://ru.wikibooks.org/wiki/Алгоритм_вычисления_дня_недели)

4 Григорианский календарь - [http://ru.wikipedia.org/wiki/Григорианский\\_календарь](http://ru.wikipedia.org/wiki/Григорианский_календарь)

5 Расстояние Левенштейна - [http://ru.wikipedia.org/wiki/Расстояние\\_Левенштейна](http://ru.wikipedia.org/wiki/Расстояние_Левенштейна)

**Вариант 27.** Программа принимает от пользователя строку, содержащую дату в формате ДД.ММ.ГГГГ и выводящую на экран 6-недельный календарь (столбцы либо строки — дни недели), содержащий введенную дату<sup>1</sup>. Эта дата должна быть особым образом выделена среди окружающих ее дат и не должна располагаться на заголовочной, первой и последней строке календаря при горизонтальном расположении дней (для столбцов — при вертикальном расположении). Необходимо учесть ввод некорректных значений (например, 46.14.2008 или 05.02.1918<sup>2</sup>).

---

<sup>1</sup> Пример календаря - <http://www.macmax.ru/calendar/>

<sup>2</sup> Григорианский календарь - [http://ru.wikipedia.org/wiki/Григорианский\\_календарь](http://ru.wikipedia.org/wiki/Григорианский_календарь)