Занятие 2. Программирование линейных алгоритмов

Линейный алгоритм — это алгоритм, в котором все команды (операторы) выполняются последовательно одна за другой.

В конце прошлого занятия был приведён пример линейной программы вычисления суммы и разности двух чисел. Вот ещё один.

Задача 1. Электронные часы показывают *h* часов, *m* мин. Определить их показания через *b* часов, *k* минут. (Электронные часы могут показывать время от 00:00 до 23:59 включительно.)

Решение.

Сложим минуты: $m_1 = m + k$. Может оказаться, что $m_1 \ge 60$. Поэтому выделим из этого числа величину $d = m_1 div 60$ для переноса в разряд часов, а для нахождения нового значения на табло минут найдём остаток от деления m_1 на 60. Новое значение часов будет $h_1 = (h + b + d) mod 24$. Здесь учтено, что через указанный интервал времени могут уже начаться новые сутки. Итак, получаем следующий алгоритм.

Ввести h, m, b, k m1:=m+k d:=m1 div 60 m1:=m1 mod 60 h1:=(h+b+d) mod 24 Вывести h1, m1

При этом от использования переменной d для экономии памяти можно отказаться. Тогда получим

Ввести h, m, b, k m1:=m+k h1:=(h+b+ m1 div 60) mod 24 m1:=m1 mod 60 Вывести h1, m1

Теперь составим программу. Для выбора типов данных переменных заметим, что значения всех переменных неотрицательны и не могут превышать 60. Выберем такой целочисленный тип данных, который расходует меньше всего памяти и позволяет переменным принимать значения от 0 до 59 включительно. По таблице 1 прошлого занятия это может быть byte или ShortInt. Возьмём, например, byte. Программа получается такой.

```
Program Time;
Var h, m, b, k, h1, m1 : Byte;
Begin
WriteLn('Teкущее время в часах и минутах ');
Read(h,m);
WriteLn('Промежуток в часах и минутах ');
Read(b,k);
m1:=m+k;
h1:=(h+b+m1 div 60) mod 24;
m1:=m1 mod 60;
WriteLn(h1, ':',m1)
End.
```

При этом мы не требуем, чтобы часы и минуты при выводе результатов выражались двузначными числами.

Наконец, введём программу в компьютер.

В системе программирования FreePascal

Просто запускаем FreePascal, например, используя созданный на предыдущем занятии в вашей папке файл fp.bat, и вводим приведённый выше текст программы. Для сохранения файла с текстом программы нажимаем F2. Это обязательно нужно сделать, чтобы не потерять результаты работы в случае зависания компьютера. Свои файлы сохраняйте в свои рабочие папки и никогда в папки самой системы FreePascal. Имя файла давайте совпадающим с именем программы, т.е. в нашем случае time.pas. Для компиляции и запуска программы на выполнение нажимаем Ctrl-F9. Если в тексте программы нет ошибок, начнётся её выполнение. Чтобы просмотреть выведенный на экран результат, нажмите Alt-F5. Для возврата в IDE нажимаем Esc. Можно было нажать и любую другую клавишу. Просто нажатие Esc не может испортить текст программы. Если нужно просто создать исполняемый файл программы (exe-файл) без запуска его на выполнение, тогда нажимаем F9 вместо Ctrl-F9. Для выхода из IDE FreePascal нажмите Alt-X. В своей папке вы найдёте файл time.exe. Это исполняемый файл этой программы.

Мне представляется весьма удобным запуск FreePascal через Far Manager. Даже в Windows-7 в этом случае удаётся¹ видеть окно программ, развёрнутое во весь экран. После закрытия FreePascal вы сможете в своей рабочей папке найти созданный вами ехе-файл и запустить его на выполнение. После окончания работы программы панели Far Manager скроют от вас результат её работы, но их можно убрать нажатием Ctrl-O. Повторное нажатие этой же комбинации клавиш вернёт панели на место.

В системе программирования Delphi

Дальше, говоря о Delphi, будем иметь в виду Delphi for Microsoft Win32 из состава Borland Developer Studio 2006. Если у вас другая версия этой системы программирования, то вы сможете разобраться с ней по аналогии со следующим описанием. Итак,

- 1. запускаем Delphi;
- 2. в меню выбираем File New Other... или просто нажимаем кнопку "New Project";
- в окне "New Items" устанавливаем Item Categories в Delphi Projects, а в правом окне выбираем Console Application и нажимаем кнопку "ОК". После этого автоматически создастся несколько строк кода будущей программы.
- 4. В меню выбираем File Save All или соответствующую кнопку на панели



инструментов. Создаём *новую папку* и *в ней* сохраняем файл проекта с именем test. По умолчанию все проекты создаются в папке Borland Studio Projects, вложенной в «Мои документы». В папке проекта сразу создаётся несколько файлов. Текст консольной программы будет храниться в файле с расширением dpr, но файлом проекта будет файл с расширением bdsproj, и если нам в дальнейшем потребуется усовершенствовать проект, то открывать мы будем именно ero².

- 5. В окне редактора вводим текст программы. При этом строку{ *тодо -oUser -cConsole Main : Insert code here* } можно просто удалить. Вывода на экран русских текстов следует избегать. Программы в Delphi разрабатываются в кодировке Windows, а при выводе на экран консольные приложения используют кодировку DOS. Из-за различной кодировки символов в DOS и Windows текст окажется нечитаемым. Поэтому все сообщения программы пишите английскими буквами.
- 6. Для компиляции программы нажимаем Ctrl-F9. Если ошибок при компиляции обнаружено не будет, в папке проекта появится файл test.exe. Можно его запускать на выполнение. Опять-таки советую делать это через Far Manager. Можно было бы нажатием клавиши F9 запустить программу на компиляцию и выполнение, но в этом случае вы не увидите результатов её работы. Обратите внимание, что назначение клавиши F9 и клавиатурной комбинации Ctrl-F9 в Delphi и FreePascal прямо противоположное.

¹ После нажатия Alt-F9 в Far Manager

 $^{^2}$ В Delphi7 файлом проекта будет файл с расширением dpr, и открывать надо будет его.

Некоторые стандартные функции языка Паскаль

Таблица 1

| Функция | Описание |
|-----------|--|
| abs(x) | Вычисление модуля числа x. X может быть типа Integer, Int64 или Real. |
| | Тип результата совпадает с типом аргумента. |
| sqrt(x) | Вычисление квадратного корня из х. Результат — вещественное число (тип |
| | Extended). |
| sqr(x) | Вычисление x^2 . Х типа Real, результат типа Extended. |
| Round(x) | Округление числа x типа Real до целого. Результат имеет тип Int64. |
| | Например, |
| | round(100.25) даёт 100, |
| | round(100.75) даёт 101, |
| | round(-100.25) даёт -100, |
| | round(-100.75) даёт -101 |
| Trunc(x) | Обрезание дробной части вещественного числа <i>x</i> . Тип результата — Int64. |
| | Например, |
| | trunc(100.25) даёт 100, |
| | trunc (100.75) даёт 100, |
| | trunc (-100.25) даёт -100, |
| | trunc (-100.75) даёт -100 |
| Pi | Возвращает 3.1415926535897932385 |
| sin(x) | Вычисление тригонометрических функций числа х. Х задаётся в радианах. |
| $\cos(x)$ | Аргумент и результат имеют тип Extended. |
| ArcTan(x) | Вычисление арктангенса х. Результат получается в радианах. Аргумент и |
| | результат имеют тип Extended. |
| exp(x) | Вычисление <i>e^x</i> |
| $\ln(x)$ | Вычисление натурального логарифма. Аргумент и результат имеют тип |
| | Extended. |

Все типы данных указаны для Delphi. Вспомните, что FreePascal может работать в режиме совместимости с Delphi.

Для иллюстрации использования этих функций решим следующую задачу.

Задача 2. Треугольник задан длинами своих сторон *a*, *b*, *c*. Найти его площадь. Длины сторон — вещественные числа.

Алгоритм решения этой задачи очень прост. Формула Герона $S = \sqrt{p(p-a)(p-b)(p-c)}$, где *p* — полупериметр, даёт требуемый результат.

Получаем программу на FreePascal.

```
{$mode Delphi}
Program Geron;
Var a,b,c,p,S : Real;
Begin
Write('a,b,c = ');
Read(a,b,c);
p := (a+b+c) / 2;
S := sqrt(p*(p-a)*(p-b)*(p-c));
WriteLn('S=',S:5:2)
End.
```

Здесь первая строка переключает FreePascal в режим совместимости с Delphi. Это сделано в данном случае исключительно в целях иллюстрации. Программа работала бы и без этого. Если эту же программу создавать в Delphi, то указанную строку писать не нужно. При выводе на экран значение площади округляется до двух знаков после запятой.

Задачи для самостоятельного решения.

Составьте программы для решения следующих задач.

- 1. Найти периметр и площадь прямоугольного треугольника по заданным длинам двух катетов *a* и *b*. (Длины катетов задаются натуральными числами, не превышающими 30000.)
- 2. Найти произведение цифр заданного трёхзначного натурального числа. (Подсказка. Вспомните про div и mod.)
- 3. На автомобиль массой *x* т *y* кг погрузили груз массой *p* т *q* кг. Найти массу автомобиля вместе с грузом в тоннах и килограммах. При решении задачи используйте только целочисленные переменные.
- 4. Дано четырёхзначное число. Найти число, образуемое при перестановке первой и второй, а также третьей и четвёртой цифр заданного числа. Например, из числа 5434 получить 4543, а из числа 7048 — число 784.
- 5. С начала суток часовая стрелка повернулась на *у* градусов (0 ≤ *y* < 360, *y* вещественное число). Определить число полных часов и число полных минут, прошедших с начала суток.