Санкт-Петербургский государственный
политехнический университет

Институт информационных технологий и управления

Кафедра компьютерных систем и программных технологий

Сиднев А.Г.

Методические указания к лабораторным работам по теме «Построение генераторов случайных величин в среде моделирования GPSS World»

Санкт-Петербург
2014

Содержание

1. Теоретические основы построения программных генераторов случайных величин

1.1. Базовая последовательность случайных чисел, используемая для формирования случайных факторов в имитационных моделях

Базовой последовательностью случайных чисел, используемой для формирования в ЭВМ случайных элементов различной природы с различными законами распределения, является совокупность случайных чисел с равномерным законом распределения:

$$f(x) = \begin{cases} \dfrac{1}{b-a}, & x \in [a,b] \\ 0, & x < a, \quad x > b \end{cases} , \tag{1.1}$$

где f (x) – дифференциальный закон распределения равномерно распределенных чисел x в интервале [a, b].

Строго говоря, на ЭВМ получить последовательность непрерывных случайных величин с равномерным законом распределения не представляется возможным. Поэтому, если считать, что число разрядов ЭВМ равно k, а случайное число сформировано согласно формуле:

$$\alpha = \sum_{j=0}^{k-1} \alpha_j 2^j , \tag{1.2}$$

где $\alpha_j = \begin{cases} 0, P_j = 1/2 \\ 1, P_j = 1/2 \end{cases}$, то $x = \alpha /(2^k - 1)$ принимает значение $i/(2^k - 1),\ i = 0, 1, 2, \dots, 2^k - 1$ с вероятностью $P_i = 1/2^k$. Такое распределение чисел называется квазиравномерным в интервале [0,1], причем математическое ожидание и дисперсия определяются следующими соотношениями:

$$\begin{aligned} M[x] &= \sum_{i=0}^{2^k-1} \frac{i}{2^k-1} \frac{1}{2^k} = \frac{1}{2}, \\ D[x] &= \sum_{i=0}^{2^k-1} \left(\frac{i}{2^k-1} - \frac{1}{2} \right)^2 \frac{1}{2^k} = \frac{1}{12} \frac{2^k+1}{2^k-1}. \end{aligned} \tag{1.3}$$

Из формул (1.2) видно, что математическое ожидание $M[x]$ точно совпадает с генеральным средним для равномерного распределения в интервале [0,1]; а дисперсия при $k \to \infty$ ассимптотически стремится к дисперсии для равномерного распределения при $a = 0, b = 1$, равной 1/12. Практически при $k > 15$ достигается необходимая точность моделирования.

1.2. Моделирование дискретных случайных величин

Пусть рассматривается случайная величина X, принимающая n исходов: $X = x_i, i = \overline{1,n}$ с вероятностями $P(X = x_i) = p_i$, $\sum_{i=1}^{n} p_i = 1$

$\sum_{i=1}^{n} p_i = 1$.

Общий метод моделирования дискретной случайной величины основан на следующем очевидном соотношении:

$$P\left(\sum_{j=1}^{i-1} p_j \le y \le \sum_{j=1}^{i} p_j \right) = p_i \,, \qquad (1.4)$$

где $p_i = P(X = x_i), i = \overline{1,n}$, а y – равномерно распределенная случайная величина в интервале $[0,1]$.

В дальнейшем используется запись $y \in R(0,1)$. Для частного случая $n = 2$ при выполнении условия $y < p_1$, $X = x_1$, в противном случае $X = x_2$, $p_1 + p_2 = 1$.

Наиболее важными дискретными случайными величинами являются целочисленные с распределением

$$p_i = P(X = i), i = 1, 2, \ldots,$$

которое можно представить простыми рекуррентными формулами:

$$p_{i+1} = p_i r(i).$$

Для таких случайных величин значение p_i и x_i – можно не записывать в памяти ЭВМ, а моделирование осуществлять по схеме, изображенной на рис. 1.1 [1].

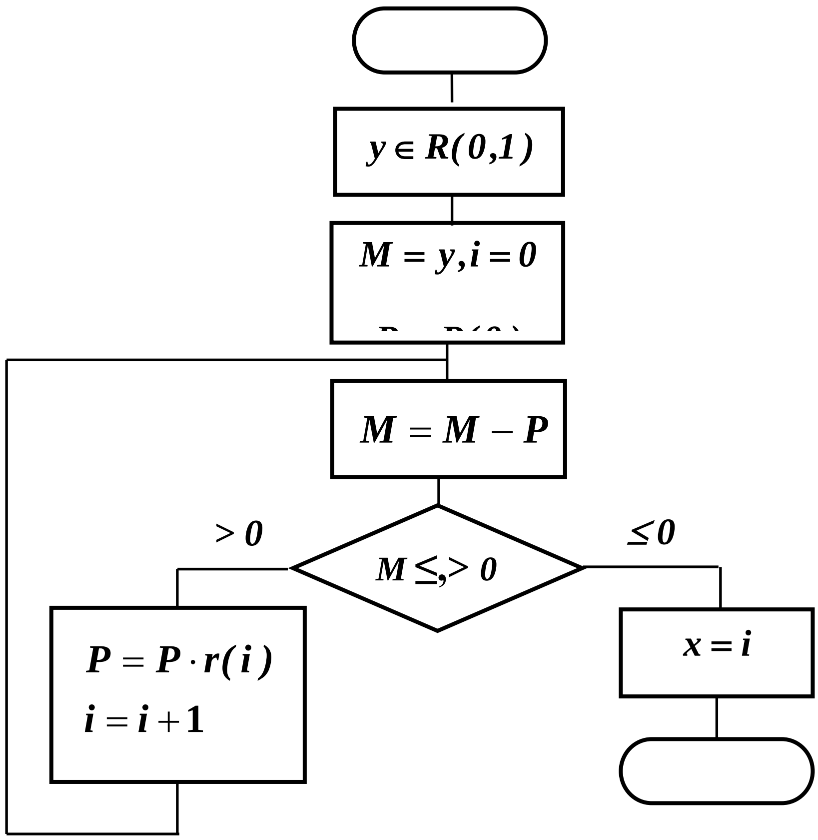

Рис. 1.1

Получим значение $r(i)$ для наиболее характерных распределений дискретных случайных величин:

1. Биномиальное распределение:

$$p_i = P(x = i) = C_n^i p^i (1 - p)^{n-i},$$

$$r(i) = \frac{p_{i+1}}{p_i} = \frac{(n-i)p}{(i+1)(1-p)}, \; i = 0, 1, \ldots, n.$$

2. Распределение Пуассона с параметром λ:

$$p_i = \frac{\lambda^i}{i!} e^{-\lambda}, \quad r(i) = \frac{\lambda}{i+1}, \; i = 0, 1, 2, \ldots$$

3. Геометрическое распределение с параметром p:

$$p_i = p(1-p)^{i-1}, r(i) = 1-p, \; i = 0,1,2, \ldots$$

Численные характеристики случайных величин с рассматриваемыми законами распределения приведены в табл. 1.

Таблица 1

Название закона распределения		Вид $P(k) / f(x)$	Среднее	Дисперсия
Дискретные	Биномиальный	$C_n^k p^k (1-p)^{n-k}$, $k = 0,1,2, \ldots, n$	np	$np(1-p)$
	Пуассона	$\frac{\lambda^k}{k!} e^{-\lambda}$	λ	λ
	Равномерный	$1/n$, $k = 0,1,2, \ldots, n$	$(n+1)/2$	$(n^2-1)/12$
Непрерывные	Равномерный	$1/(b-a)$	$(a+b)/2$	$(b-a)^2/12$
	Нормальный	$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$	μ	σ^2
	Экспоненциальный	$\lambda e^{-\lambda x}$	$1/\lambda$	$(1/\lambda)^2$

1.3. Моделирование непрерывных случайных величин

Рассмотрим сначала общие приемы получения случайных чисел с заданным законом распределения из равномерно распределенных случайных чисел.

Метод обратного преобразования

Пусть y – случайная величина, равномерно распределенная на отрезке $[0,1]$, тогда случайная величина x, являющаяся решением уравнения:

$$y = \int_{-\infty}^{x} f(z)dz \qquad (1.5)$$

имеет плотность распределения $f(x)$.

Обоснование метода. Если $f(x)$ – плотность распределения, то $y = F(x)$, где $F(x)$ – интегральная функция распределения, монотонно возрастающая и непрерывная функция. Тогда в соответствии с [2]:

$$P[y < F^{-1}(y)] = F[F^{-1}(y)] = y, \qquad (1.6)$$

что свидетельствует о равномерном законе распределения y:

$$P[y_1 < y < y_2] = y_2 - y_1,$$

т.е. вероятность попадания случайной величины y в интервал $[y_1, y_2]$ равна длине интервала.

Рассмотрим примеры.

Пример 1

Необходимо сгенерировать последовательность случайных чисел, равномерно распределенных на интервале $[a,b]$. Используя (1.5), получим:

$$y_i = \int_a^{x_i} f(z)\,dz = \int_a^{x_i} \frac{1}{b-a} dz = \frac{x_i - a}{b-a}, \qquad (1.7)$$

$$x_i = a + y_i(b-a).$$

Пример 2

Необходимо получить последовательность случайных чисел, имеющих экспоненциальный закон распределения:

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, & 0 \le x < \infty \\ 0, & x < 0 \end{cases}. \tag{1.8}$$

В соответствии с (1.5) имеем:

$$y_i = \int_0^{x_i} \lambda e^{-\lambda z} dz = 1 - e^{-\lambda x_i} \text{, откуда}$$

$$x_i = -\frac{1}{\lambda} \cdot ln(1 - y_i). \tag{1.9}$$

Так как величина $(1 - y_i)$ так же как и y_i имеет равномерное распределение на интервале $[0,1]$, то формула (1.9) может быть записана другим способом: $x_i = -\frac{1}{\lambda} \cdot ln\, y_i$.

Пример 3

Необходимо смоделировать последовательность случайных чисел, имеющих закон распределения Вейбула:

$$f(x) = \lambda \alpha x^{\alpha - 1} e^{-\lambda x^{\alpha}}, \quad \lambda > 0, \alpha > 0, \quad 0 < x < \infty, .$$

В соответствии с (1.3) имеем:

$$y_i = \int_0^{x_i} \lambda \alpha z^{\alpha - 1} e^{-\lambda z^{\alpha}} dz = 1 - e^{-\lambda x_i^{\alpha}} \text{, откуда}$$

$$x_i = \left[-\frac{1}{\lambda} ln(1 - y_i) \right]^{\frac{1}{\alpha}}. \tag{1.10}$$

При $\alpha = 1$ получаем соотношение (1.9).

Пример 4

Необходимо смоделировать случайную величину с релеевским законом распределения с плотностью распределения:

$$f(x) = \frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}}, \; x \ge 0. \tag{1.11}$$

Используя (6.8) получим:

$$y_i = \int_0^{x_i} \frac{z}{\sigma^2} \cdot e^{\frac{-z^2}{2\sigma^2}} dz = 1 - e^{\frac{-x_i^2}{2\sigma^2}},$$

откуда имеем:

$$x_i = \sigma\sqrt{-2\ln(1-y_i)} = \sigma\sqrt{-2\ln y_i}\,. \tag{1.12}$$

Общий принцип генерации случайных чисел в соответствии с заданным методом включает:

1. Выборку числа $y_i \in R(0,1)$.

2. Преобразование в соответствии с соотношением (1.5).

К сожалению, не всегда существуют элементарные преобразования равномерно распределенных чисел для получения случайных величин с заданным законом распределения. В частности, у случайных величин с нормальным распределением функция $F^{-1}(y_i)$ не выражается в явном виде через элементарные функции. В этих случаях для формирования случайных величин с заданным распределением используются различные аппроксимации $F^{-1}(y_i)$.

Использование предельных теорем

Для имитации определенных законов распределения используют предельные теоремы теории вероятностей. Так, например, для получения нормального закона распределения используется свойство асимптотической сходимости сумм независимых случайных величин к нормальному закону распределения.

Если взять n значений чисел из совокупности, равномерно распределенной на интервале $[0,1]$ [1], то при $n = 12$ величину

$$x = \sum_{i=1}^{12} y_i - 6 \qquad (1.13)$$

можно считать распределенной по нормальному закону с параметрами

$$M[x] = 0, \; D[x] = 1, \text{ т.е. } x = N(0,1), \text{а } f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}.$$

Для получения нормального закона распределения $N(\mu,\sigma)$ с плотностью

$$f(x') = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-1}{2}\frac{(x'-\mu)^2}{\sigma^2}}$$

используют линейное преобразование:

$$x' = \mu + \sigma \cdot x.$$

Рассмотрим другие возможные подходы для генерирования нормального распределения [2].

1.4. Некоторые специальные методы моделирования случайных величин

Метод Бокса и Маллера

Согласно этому методу генерируется пара нормированных нормальных чисел x_1 и x_2, имеющих закон распределения $N(0,1)$ из двух стандартных случайных равномерно распределенных на интервале $[0,1]$ чисел y_1, y_2 с помощью следующих формул:

$$\begin{aligned} x_1 &= \sqrt{-2\ln y_1}\,\sin 2\pi y_2, \\ x_2 &= \sqrt{-2\ln y_1}\,\cos 2\pi y_2. \end{aligned} \qquad (1.14)$$

Используя преобразование (1.14) в виде:

[1] Дисперсия равномерно распределенной в интервале $[a,b]$ величины равна $\frac{(b-a)^2}{12}$.

$$y_1 = exp\,[-\frac{1}{2}(x_1^2 + x_2^2)]$$
$$y_2 = \frac{1}{2\pi} arctg\,\frac{x_1}{x_2}, \tag{1.15}$$

а также известные соотношения для получения законов распределения от функций случайных величин [2]:

$$f(x_1, x_2) = f(y_1, y_2)\frac{d(y_1, y_2)}{d(x_1, x_2)}, \tag{1.16}$$

где $\frac{d(y_1, y_2)}{d(x_1, x_2)}$ — якобиан преобразования.

Следует напомнить, что Якобиан (определитель Якоби, функциональный определитель) есть определитель матрицы Якоби:

$$det\begin{pmatrix} \frac{\partial u_1}{\partial x_1}(x) & \frac{\partial u_1}{\partial x_2}(x) & ... & \frac{\partial u_1}{\partial x_n}(x) \\ \frac{\partial u_2}{\partial x_1}(x) & \frac{\partial u_2}{\partial x_2}(x) & ... & \frac{\partial u_2}{\partial x_n}(x) \\ ... & ... & ... & \\ \frac{\partial u_n}{\partial x_1}(x) & \frac{\partial u_n}{\partial x_2}(x) & ... & \frac{\partial u_n}{\partial x_n}(x) \end{pmatrix}, \text{ где}$$

$u:\ R^n \rightarrow R^n$ — векторная функция:
$u = (u_1, \ ..., u_n),\quad u_i = u_i(x_1, \ ..., x_n),\ i = 1, ..., n$.

В данном случае $u = (u_1, \ ..., u_n) = (y_1, y_2)$,

причем совместная плотность распределения системы независимых случайных величин (y_1, y_2)

$$f(y_1, y_2) = f(y_1) f(y_2) = 1, \qquad y_1, y_2 \in R(0,1)$$

Поэтому с учетом (1.16)

$$f(x_1, x_2) = \frac{1}{2\pi} e^{-\frac{1}{2}(x_1^2 + x_2^2)}. \tag{1.17}$$

Соотношение (1.17) свидетельствует о том, что числа x_1 и x_2 являются независимыми, имеющими закон $N(0,1)$.

Метод Марсальи-Брея

Ими предлагается модификация метода Бокса и Маллера, столь же легкая для программирования и дающая точные результаты, но дающая их быстрее.

По этому методу:

1. Генерируются два случайных числа y_1, y_2.

2. Проводится их центрирование, т.е. приведение к нулевому математическому ожиданию:

$$v_1 = -1 + 2y_1; \quad v_2 = -1 + 2y_2 \ .$$

3. Вычисляется $s = v_1^2 + v_2^2$, при $s > 1$ начинаем цикл снова, т.е. возвращаемся к п.1, в случае $s \le 1$ вычисляем:

$$x_1 = v_1\sqrt{\frac{-2\ln s}{s}}, \quad x_2 = v_2\sqrt{\frac{-2\ln s}{s}}. \tag{1.18}$$

Модификация Марсальи-Брея позволяет исключить необходимость вычисления синусов и косинусов, хотя для квадратных корней и натуральных логарифмов такая необходимость сохраняется.

В отличие от соотношения (1.13) соотношения (1.14) и (1.18) являются точными, т.е. дают строго нормальный закон распределения по выбранным равномерно распределенным числам. Кроме того, генератор, построенный в соответствии с соотношением (1.13), имеет ряд недостатков.

1. Для вычисления каждой нормально распределенной величины ему необходимо 12 равномерно распределенных случайных чисел. В силу наличия периода у любого генератора $R(0,1)$ при необходимости получения большого количества случайных переменных, имеющихся в распоряжении равномерно распределенных чисел может не хватить.

2. При генерировании “хвостов” нормального распределения этот метод работает плохо. Как следует из [2], за пределами порогов 2σ имеет место сильное расхождение характеристик этого генератора с желаемыми.

Метод Неймана (метод исключения)

Данный метод является достаточно универсальным для моделирования случайных величин, возможные значения которых не выходят за пределы некоторого ограниченного интервала $[a,b]$ (речь идет об усеченных законах распределения). Общий принцип генерации числа с плотностью $f(x)$ распределения вероятностей следующий:

1. Из генератора $R(0,1)$ выбираются два числа y_1, y_2.

2. Проводятся преобразования:

$$u_1 = a + (b - a)y_1, \quad u_2 = f_M \cdot y_2,$$

где $f_M = max\, f(x), \; x \in [a,b]$.

3. В качестве реализации случайной величины x берется число u_1, удовлетворяющее условию $u_2 < f(u_1)$. Пары чисел y_1, y_2, не удовлетворяющие данному неравенству, отбрасываются. См. рис. 1.2.

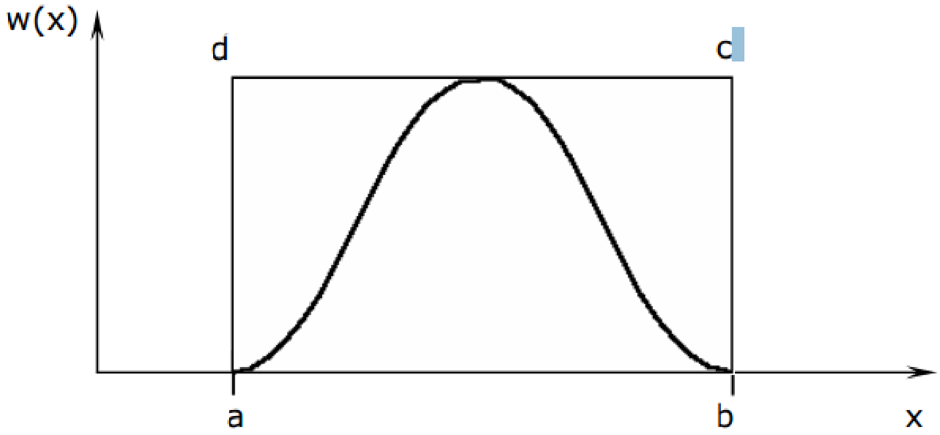

Рис. 1.2. К методу Неймана.

Метод суперпозиции

При известных условной плотности распределения $f(x/z)$ и плотности $f(z)$ параметра z плотность распределения $f(x)$ выражается следующим образом:

$$f(x) = \int_{-\infty}^{+\infty} f(x/z)\, f(z) dz. \tag{1.19}$$

Если параметр z принимает только целые значения $z_i, i = 1,2,\ldots,n$ с вероятностями p_i, $\sum_{i=1}^{n} p_i = 1$, то

$$f(x) = \sum_{i=1}^{n} p_i f_i(x), \tag{1.20}$$

где $p_i = P(z = z_i)$, а $f_i(x) = f(x / z_i)$.

В случае $n = 2$ и нормального закона распределения соотношение (1.20) выглядит следующим образом:

$$f(x) = p_1 \frac{1}{\sigma_1\sqrt{2\pi}} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}} + p_2 \frac{1}{\sigma_2\sqrt{2\pi}} e^{-\frac{(x-\mu_2)^2}{2\sigma_2^2}}, \qquad (1.21)$$

$$p_1 + p_2 = 1.$$

При генерации распределения (1.21) сначала реализуется выбор дискретной случайной величины в соответствии с соотношением (1.4), а далее с учетом полученного значения z генерируется число с нормальным законом распределения и параметрами μ_i, σ_i, $i = 1,2$ согласно соотношениям (1.13), (1.14), (1.18).

Метод кусочной аппроксимации

Среди различных приближенных приемов моделирования случайных величин наиболее простым и достаточно универсальным является метод кусочной аппроксимации, предложенный Н.П. Бусленко [3].

Сущность этого метода состоит в следующем. Пусть требуется получить случайную величину x с функцией плотности $f(x)$. Предположим, что область возможных значений величины x ограничена интервалом $[a,b]$. Разобьем интервал $[a,b]$ на несколько достаточно малых интервалов $[a_i, a_{i+1}]$, $i = \overline{0, n-1}$, $a_0 = a, a_n = b$ так, чтобы распределение заданной случайной величины x в пределах этих интервалов можно было достаточно точно аппроксимировать каким-нибудь простым распределением, например, равномерным.

Пусть p_i – вероятность попадания случайной величины в каждый из интервалов $[a_i, a_{i+1}]$. Обычно принимают, $p_i = 1/n$, т.е.

$$p_i = \int_{a_i}^{a_{i+1}} f(x)dx = \frac{1}{n}. \qquad (1.22)$$

Исходя из соотношения (1.22), предварительно для заданной $f(x)$ рассчитывают границы интервалов $a_i, i = \overline{0, n-1}$.

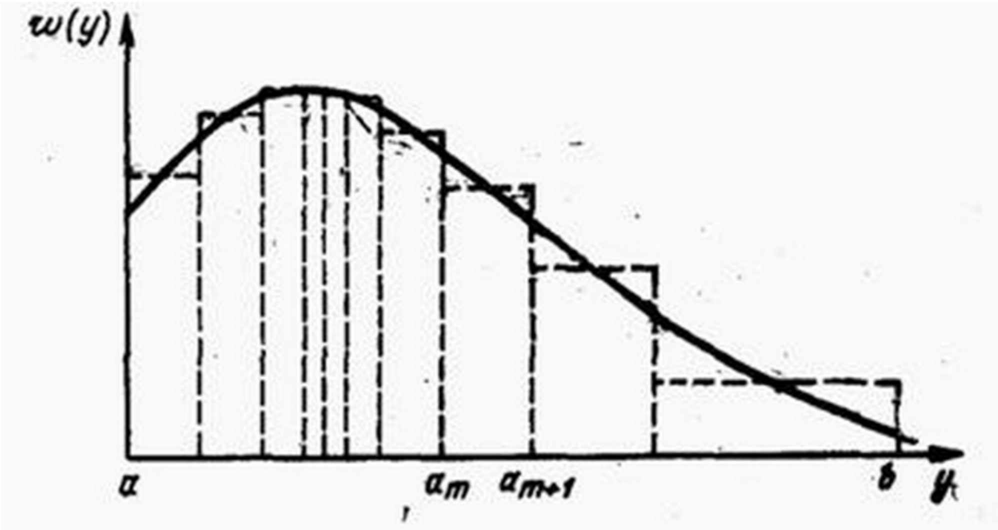

Рис. 1.3

В соответствии с выбранной аппроксимацией приходят к следующему способу преобразования равномерно распределенных чисел в случайные числа с заданным законом распределения $f(x)$:

1. Из генератора $R(0,1)$ выбираются пара случайных чисел y_1, y_2.
2. По первому числу y_1 в соответствии с соотношением (1.4), где значения $p_i = 1/n$, выбирается интервал $[a_i, a_{i+1}]$.
3. По второму числу y_2 рассчитывается величина x:
   $x = a_i + y_2(a_{i+1} - a_i)$,имеющая заданную плотность $f(x)$.

Такой алгоритм (рис. 1.3.) является довольно экономичным по количеству требуемых операций, которое не зависит от числа n, т.е. не зависит от точности аппроксимации. Однако с увеличением точности аппроксимации возрастает количество ячеек памяти, требуемых для хранения величин a_i, $i = \overline{0,n}$, что является недостатком рассмотренного метода, в особенности при больших n.

2. Методика построения программ генерации случайных чисел в среде GPSS World

Методика построения ГСЧ в среде GPSS World опирается на приведенные выше теоретические предпосылки, в частности, на обратную теорему, а также на широкое использование выразительных возможностей синтаксиса языка GPSS.

Ниже приводятся комментированные GPSS-программы, обеспечивающие на выходе получение случайных величин, распределенных по закону, близкому е нормальному. В большинстве предложенных программ используется предельная центральная теорема. Степень соответствия программного генератора эталонному закону оценивается с использованием пакета STATGRAPHICS и Optimization Toolbox пакета Matlab.

<u>Задание 1</u>: Моделирование нормального закона распределения путем суммирования N_1 независимых случайных величин, распределенных по показательному (экспоненциальному) закону распределения (с использованием результатов центральной предельной теоремы).

Текст GPSS-программы Norm-Exp.gps

```
* Моделирование нормального закона распределения путем суммирования N_1
* НСВ, распределенных по показательному (экспоненциальному) закону
распределения

* Инициализация исходных данных, переменных, функций и выражений

N_1      EQU       10
N_2      EQU       5
ROW      VARIABLE  1001-X2
E_XP     VARIABLE  (Exponential(1,0,10))

RAND1    MATRIX    ,1000,1
         GENERATE  ,,,1000
         SAVEVALUE 2+,1
         ASSIGN    1,N_1

LOOP1    SAVEVALUE 1+,V$E_XP
         LOOP      1,LOOP1
         MSAVEVALUE                 RAND1,V$ROW,1,X1
         SAVEVALUE 1,0
         TERMINATE 1
```

<u>Задание 2</u>: Моделирование нормального закона распределения путем суммирования N_1 независимых случайных величин, являющимися суммами N_2 равномерно распределенных (R(0,1)) независимых случайных величин
(с использованием результатов центральной предельной теоремы).

Текст GPSS-программы Norm-Uniform.gps

```
* Моделирование нормального закона распределения путем суммирования N_1
* НСВ, являющимися суммами N_2 равномерно распределенных (R(0,1))
независимых
* случайных величин


* Инициализация исходных данных, переменных, функций и выражений

N_1      EQU       5
N_2      EQU       12
ROW      VARIABLE  1001-X2
UNIF     FUNCTION  RN1,C2
         0.0,0.0/1.0,1.0

RAND1    MATRIX    ,1000,1
         GENERATE  ,,,1000
         SAVEVALUE 2+,1
```

```
          ASSIGN    1,N_1
LOOP2     ASSIGN    2,N_2
LOOP1     SAVEVALUE 1+,FN$UNIF
          LOOP      2,LOOP1
          SAVEVALUE 1-,(N_2/2)
          SAVEVALUE 3+,X1
          SAVEVALUE 1,0
          LOOP      1,LOOP2
          MSAVEVALUE                RAND1,V$ROW,1,X3
          SAVEVALUE 3,0
          TERMINATE 1
```

Задание 3: Моделирование нормального закона распределения путем суммирования N_1 независимых случайных величин, распределенных по гиперэкспоненциальному закону (с использованием результатов центральной предельной теоремы).

Текст GPSS-программы Norm-HyperExp.gps

```
* Моделирование нормального закона распределения путем суммирования N_1
* НСВ, распределенными по гиперэкспоненциальному закону


* Инициализация исходных данных, переменных, функций и выражений
          RMULT     555,37

N_1       EQU       12
E_XP1     VARIABLE  (Exponential(1,0,10))
E_XP2     VARIABLE  (Exponential(1,0,20))
ROW       VARIABLE  1001-X2

RAND1     MATRIX    ,1000,1
          GENERATE  ,,,1000
          SAVEVALUE 2+,1
          ASSIGN    1,N_1
LOOP1     TRANSFER  .5,E_1,E_2
E_1       SAVEVALUE 1+,V$E_XP1
          TRANSFER  ,NEXT
E_2       SAVEVALUE 1+,V$E_XP2
NEXT      LOOP      1,LOOP1

          MSAVEVALUE                RAND1,V$ROW,1,X1
          SAVEVALUE 1,0
          TERMINATE 1
```

Задание 4: Моделирование нормального распределения путем суммирования N_1 независимых случайных величин, распределенных по закону Эрланга k-го порядка (k=N_2) (с использованием результатов центральной предельной теоремы). Распределение Эрланга моделируется посредством суммирования N_2 независимых случайных величин, распределенных по показательному (экспоненциальному) закону.

Текст GPSS-программы Norm-Erlang.gps

```
* Моделирование нормального распределения путем суммирования N_1
* НСВ, распределенных по закону Эрланга k-го порядка (k=N2).
* Распределение Эрланга моделируется посредством суммирования N2
```

```
* НСВ, распределенных по показательному (экспоненциальному) закону

* Инициализация исходных данных, переменных, функций и выражений

N_1      EQU       10
N_2      EQU       5
ROW      VARIABLE  1001-X2
E_XP     VARIABLE  (Exponential(1,0,10))

RAND1    MATRIX    ,1000,1
         GENERATE  ,,,1000
         SAVEVALUE 2+,1
         ASSIGN    1,N_1
LOOP2    ASSIGN    2,N_2
LOOP1    SAVEVALUE 1+,V$E_XP
         LOOP      2,LOOP1
         SAVEVALUE 3+,X1
         SAVEVALUE 1,0
         LOOP      1,LOOP2
         MSAVEVALUE                RAND1,V$ROW,1,X3
         SAVEVALUE 3,0
         TERMINATE 1
```

Задание 5: Моделирование нормального распределения путем суммирования
N_1 независимых случайных величин, распределенных по нормальному же закону по методу Бокса и Малера. Так же, как и ранее используются результаты центральной предельной теоремы

Текст GPSS-программы Box-Maler.gps

```
* Моделирование нормального распределения путем суммирования
* N_1 НСВ, распределенных по нормальному же закону по методу
* Бокса и Малера


* Инициализация
         RMULT     555,37
N_1      EQU       10
N_2      VARIABLE  N_1/2
ROW      VARIABLE  1001-X2
UNIF1    FUNCTION  RN1,C2
         0.0,0.0/1.0,1.0
UNIF2    FUNCTION  RN2,C2
         0.0,0.0/1.0,1.0
IKS1     VARIABLE  SQR(-2#LOG(FN$UNIF1))#SIN(6.28#FN$UNIF2)
IKS2     VARIABLE  SQR(-2#LOG(FN$UNIF1))#COS(6.28#FN$UNIF2)
RAND1    MATRIX    ,1000,1
         GENERATE  ,,,1000
         SAVEVALUE 2+,1
         ASSIGN    1,V$N_2
LOOP1    SAVEVALUE 1+,V$IKS1
         SAVEVALUE 1+,V$IKS2
         LOOP      1,LOOP1
         MSAVEVALUE                RAND1,V$ROW,1,X1
         SAVEVALUE 1,0
         TERMINATE 1
```

Задание 6: Моделирование нормального распределения путем суммирования
N_1 независимых случайных величин, распределенных по нормальному же закону по методу Марсальи-Брея. Так же, как и ранее используются результаты центральной предельной теоремы

Текст GPSS-программы Marsaleuy-Brey.gps

```
* Моделирование нормального распределения путем суммирования
* N_1 НСВ, распределенных по нормальному же закону по методу
* Марсальи-Брея


* Инициализация
          RMULT     555,37
NNN1      EQU       4
NNN2      VARIABLE  NNN1/2
ROW       VARIABLE  1001-X2
UNIF1     FUNCTION  RN1,C2
          0.0,0.0/1.0,1.0
UNIF2     FUNCTION  RN2,C2
          0.0,0.0/1.0,1.0
WV1       VARIABLE  (-1+2#FN$UNIF1)
WV2       VARIABLE  (-1+2#FN$UNIF2)
SSS       VARIABLE  X$WV1^2+X$WV2^2
IKS1      VARIABLE  X$WV1#SQR((-2#LOG(X$SSS)/X$SSS))
IKS2      VARIABLE  X$WV2#SQR((-2#LOG(X$SSS)/X$SSS))

RAND1     MATRIX    ,1000,1
          GENERATE  ,,,1000
          SAVEVALUE 2+,1
          ASSIGN    1,V$NNN2
MET1      SAVEVALUE WV1,V$WV1
          SAVEVALUE WV2,V$WV2
          SAVEVALUE SSS,V$SSS
          TEST L    X$SSS,1,MET1
          SAVEVALUE 1+,V$IKS1
          SAVEVALUE 1+,V$IKS2
          LOOP      1,MET1
          MSAVEVALUE                 RAND1,V$ROW,1,X1
          SAVEVALUE 1,0
          TERMINATE 1
```

Задание 7: Сформировать вектор значений независимых случайных величин, являющихся интервалами между событиями потока, представляющего собой сумму некоторого числа (STREAMS) релеевских потоков.[2]

---

[2] В данной программе используется линейная аппроксимация функции распределения вероятностей по закону Релея. Альтернативный вариант – использование обратной

Текст GPSS-программы Streams_Summing.gps

```
* Программа суммирования потоков. Степень
* близости суммированного потока к простейшему исследуется
* дополнительно. Близость к простейшему потоку определяется
* не только числом, но и законом распределения интервалов
* между событиями однотипных суммируемых потоков


* инициализация переменных, выражений, сохраняемых величин

STREAMS   EQU       2; задаем число суммируемых потоков
TRIES     EQU       100
PASSED    VARIABLE  N$CNT+1
I_NT      VARIABLE  AC1-X$T_IME
SROK      MATRIX    ,100,1
          INITIAL   X$T_IME,0
RELEI     FUNCTION  RN1,C50
0,0/0.05,320/0.075,395/0.1,459/0.125,516/
0.15,570/0.2,668/0.225,714/0.25,768/
0.275,802/0.3,844/0.31,861/0.325,886/
0.34,911/0.36,944/0.375,969/0.39,994/
0.4,1011/0.41,1027/0.425,1052/0.44,1077/
0.46,1110/0.475,1135/0.49,1160/0.5,1177/
0.51,1194/0.525,1220/0.54,1246/0.56,1281/
0.575,1308/0.59,1335/0.6,1354/0.625,1401/
0.65,1449/0.675,1499/0.7,1552/0.725,1607/0.75,1665/
0.775,1727/0.8,1794/0.825,1867/0.85,1948/0.875,2039/
0.9,2146/0.925,2276/0.95,2448/
0.96,2537/0.97,2648/0.99,3035/0.995,3255/
          RMULT     37
          GENERATE  ,,,STREAMS
NTR       ADVANCE   FN$RELEI
CHOSEN    MSAVEVALUE                 SROK,V$PASSED,1,V$I_NT
          SAVEVALUE T_IME,AC1
CNT       TEST GE   V$PASSED,TRIES,NTR
          TERMINATE 1
```

Задание 8: Моделирование случайной величины, распределенной по закону Эрланга k-го порядка

Распределение Эрланга имеет следующие характеристики:

$$f_\alpha(x) = \frac{\mu(\mu x)^{\alpha-1}}{(\alpha-1)!} \cdot e^{-\mu x} \qquad M[x] = \frac{\alpha}{\mu}, \quad D[x] = \frac{\alpha}{\mu^2}. \tag{1}$$

Распределение Эрланга порядка α есть частный случай Гамма-распределения.

В системе GPSS World Гамма-распределение задается следующим образом с параметрами α, β, λ, где целочисленное значение α можно интерпретировать как порядок распределения Эрланга, λ — константа, добавляемая к случайной величине (принимаем ее равной нулю), β —

теоремы в форме выражения $x_i = \sigma\sqrt{-2\ln(1-y_i)} = \sigma\sqrt{-2\ln y_i}$, см. пример 4, раздел 1.3

математическое ожидание случайной величины, распределенной по показательному закону, такой, что сумма α именно таких величин и образует случайную величину, распределенную по Эрлангу порядка α. То есть $\beta = \frac{1}{\mu}$ в формулировке (1) распределения Эрланга.

Ниже приводится фрагмент из Help GPSS World с пояснениями к Гамма-распределению, рис. 2.1.

$\alpha = Shape$

$\beta = Scale$

$\lambda = Locate$

$$f(x) = \frac{\beta^{-\alpha} (x - \lambda)^{\alpha - 1} e^{-\frac{(x-\lambda)}{\beta}}}{\Gamma(\alpha)}, \quad x > \lambda$$

$$= 0, \qquad otherwise.$$

where $\Gamma(\alpha)$ is the gamma function, defined by

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} dt.$$

$Mean: \ \alpha\beta + \lambda$

$Variance: \ \alpha\beta^2$

Рис. 2.1. Формальное задание Гамма-распределения в GPSS World

Таким образом, для генерации Эрланга, скажем, 2-го порядка с математическим ожиданием 10 следует в системе GPSS World задать распределение следующим образом:

GAMMA (1, 0, 5, 2), где 1 — номер генератора, одного из 7, случайной величины, распределенной по закону равномерной плотности в интервале от 0 до 999. 0 — нулевой сдвиг, 5 — М.О. экспоненциально распределенной случайной величины, из которой «изготовлено» распределение Эрланга 2-го порядка, 2 — порядок распределения Эланга.

Коэффициент вариации $\frac{\sigma}{M}$ для Эрланга порядка α равен $\frac{1}{\sqrt{\alpha}}$. Таким образом, при стремлении α к бесконечности, $\frac{\sigma}{M}$ стремится к 0.

Задание 9: Моделирование случайной величины, распределенной по гипергеометрическому закону

Частный случай гипергеометрического закона.

$$f(t) = 0{,}5\lambda_1 e^{-\lambda_1 t} + 0{,}5\lambda_2 e^{-\lambda_2 t}$$

Пусть $\lambda_2 = b\lambda_1$, тогда квадрат коэффициента вариации

$$C_b^2 = \left(\frac{\sigma}{M}\right)^2 = \frac{4\left[1+\frac{1}{b^2}\right]}{\left[1+\frac{1}{b}\right]^2} - 1$$

При уменьшении параметра b коэффициент вариации увеличивается, оставаясь при этом больше 1.

2.1. Запись данных в файл в среде GPSS

Ниже приводится GPSS-программа построения генератора случайных величин по заданию 3 «Моделирование нормального закона распределения путем суммирования N_1 независимых случайных величин, распределенных по гиперэкспоненциальному закону», дополненная процедурой FileWrite, записывающей формируемые программой данные в файл result.txt.

```
* Инициализация исходных данных, переменных, функций и выражений
RMULT  555,37
N_1  EQU  12
E_XP1  VARIABLE  (Exponential(1,0,10))
E_XP2  VARIABLE  (Exponential(1,0,20))
ROW  VARIABLE  1001-X2
RAND1  MATRIX  ,1000,1
GENERATE  ,,,1000
SAVEVALUE 2+,1
ASSIGN  1,N_1
LOOP1  TRANSFER  .5,E_1,E_2
E_1  SAVEVALUE 1+,V$E_XP1
TRANSFER  ,NEXT
E_2  SAVEVALUE 1+,V$E_XP2
NEXT  LOOP  1,LOOP1
MSAVEVALUE  RAND1,V$ROW,1,X1
SAVEVALUE 66,(FileWrite(X1))
SAVEVALUE 1,0
TERMINATE 1
START 1000
PROCEDURE FileWrite(NUM) BEGIN
          Open(2,"RESULT.TXT");
          Write(2, NUM);
          Close(2);
          RETURN 777;

END;
```

Фрагмент данных, записанных в файл result.txt.

```
140.8916251472999
132.7594593025391
169.1491289840554
177.6338328481335
106.5295528533588
207.1514964237383
120.7007102397784
226.033561733213
180.2706745109786
158.8007692111228
132.5520668020645
113.6141516128634
181.9646807448538
119.4240136257342
200.2003529666064
171.6196858898334
141.9015979133013
117.7921357689222
88.51291138482449
168.5206175254924
260.6581414302436
178.9567023875366
209.9085637095492
240.6952559168799
216.5142161183914
240.7389703078302
```

Для обработки данных в системе Matlab необходимо выполнить серию команд, которая приводится ниже в виде выдержки из журнала Matlab и предписывает системе открыть файл result.txt, записать данные из него в массив A, построить выборочную гистограмму и проверить гипотезу о нормальном законе распределения данных с помощью теста χ^2.

```
>> fid = fopen('result.txt');

>> A = fscanf(fid, '%f');

>> hist(A, 30);

>> x  = chi2gof(A)
```

3. Использование статистических пакетов программ для тестирования программных генераторов случайных величин

3.1. Пакет STATGRAPHICS

3.1.1. Функциональные возможности пакета STATGRAPHICS

STATGAPHICS обеспечивает широкие возможности статистической обработки данных в рамках задачи тестовой проверки программных генераторов случайных величин. Функции STATGRAPHICS представлены разделами основного меню пакета на рис. 3.1.

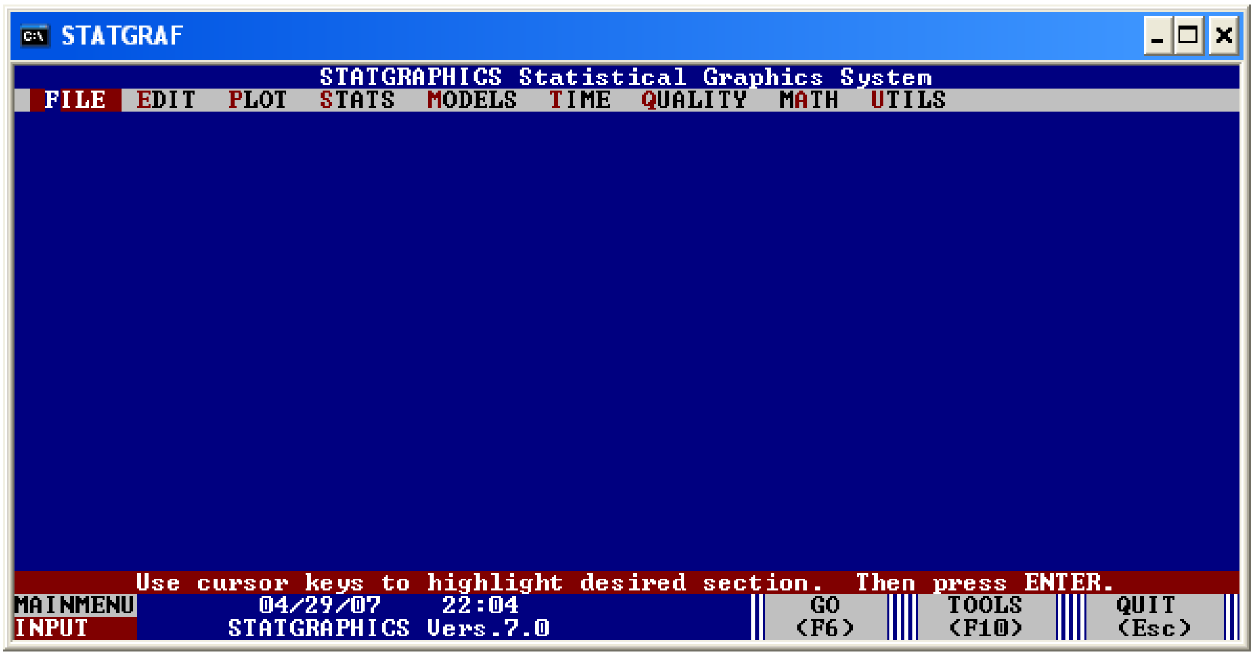

Рис. 3.1. Разделы основного меню пакета STATGRAPHICS

Каждый из основных разделов пакета STATGRAPHICS объединяет соответствующее количество тематических процедур, некоторые из которых, имеющие непосредственное отношение к теме лабораторных работ, рассматриваются ниже.

Импортирование данных в среду пакета STATGRAPHICS

Файлы с данными представляют собой записанные в столбец числа, прижатые к левому краю страницы, и должны иметь расширение «txt».

Для импортирования файла он должен быть скопирован в поддиректорий DATA директория, в который инсталлирован STATGRAPHICS (рис.3.2.), команда FILE – IMPORT DATA FILES.

Далее в открывшемся после нажатия клавиши F6 поле необходимо вписать название файла и ширину поля, отведенного под число (Field widths), рис. 3.3. После этого исследуемый файл внедряется в среду STATGRAPHICS и может быть наблюдаем, рис. 3.4 и рис. 3.5, а также подвергнут статистическому анализу.

Следует отметить, что скопированный с расширением «txt» файл порождает после выполнения команды IMPORT DATA FILES файл с расширением var1. Именно этим файлом в дальнейшем оперирует STATGRAPHICS. В частности, файл box1.txt порождает файл box1.var1, последний и необходимо указывать в качестве объекта исследования STATGRAPHICS.

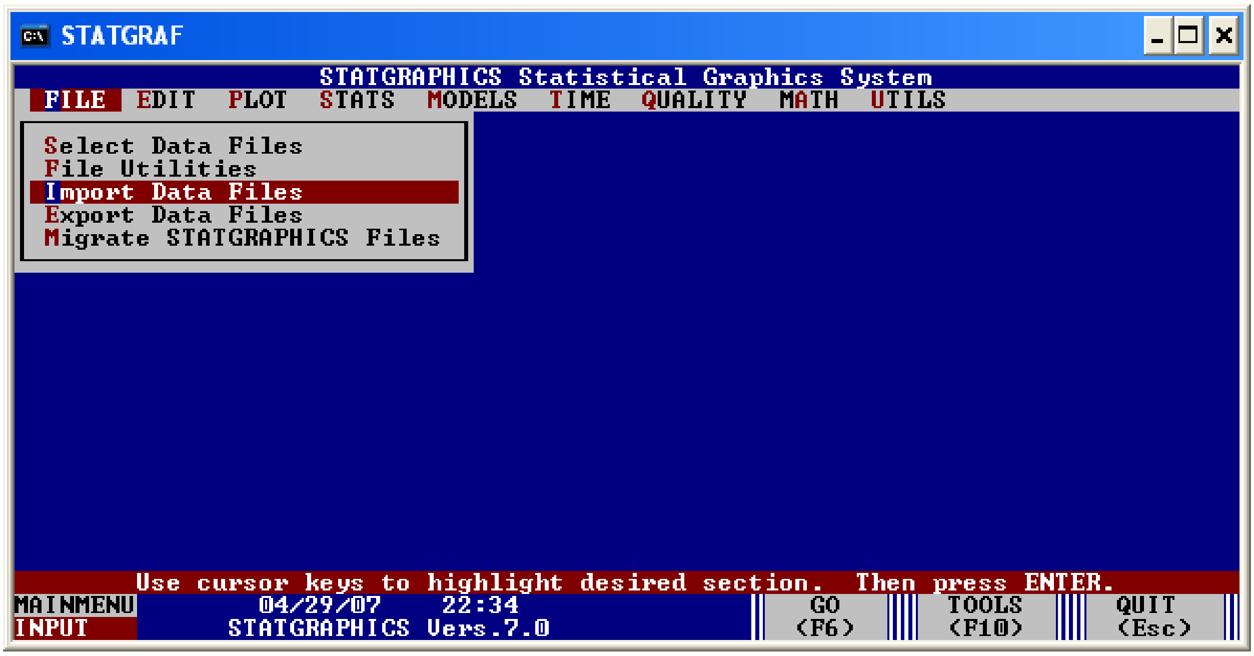

Рис. 3.2. Импортирование файла в среду STATGRAPHICS

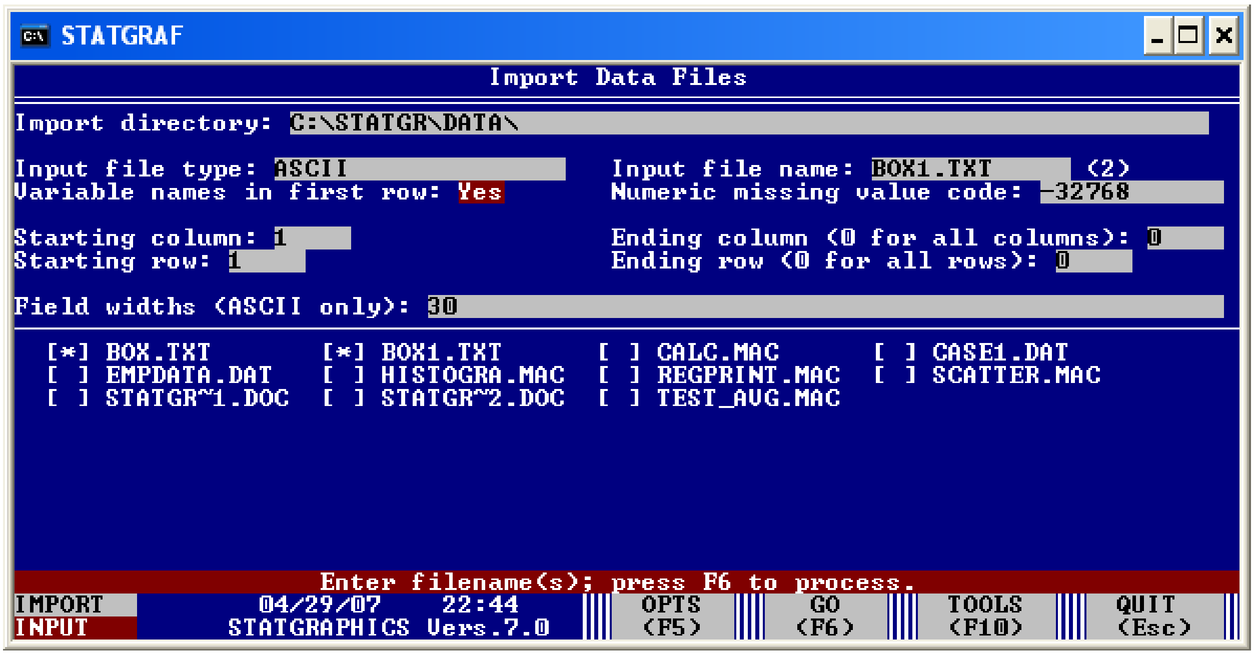

Рис. 3.3

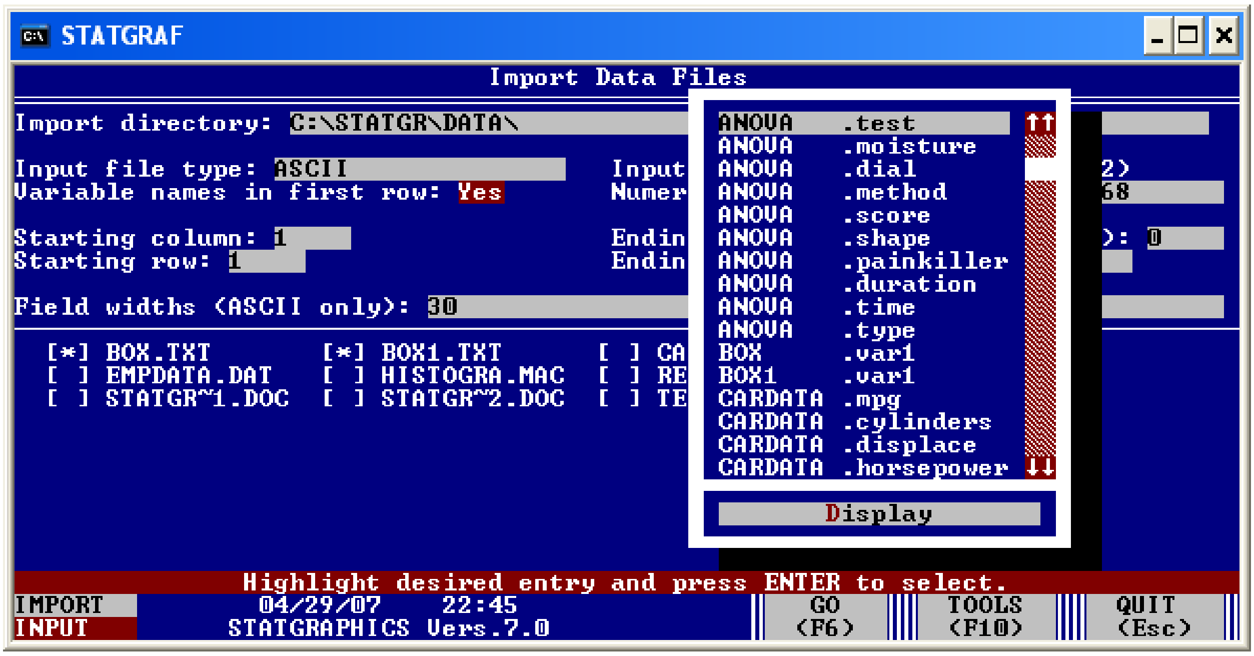

Рис. 3.4. Обзор файлов поддиректория DATA, импортированных в STATGRAPHICS с помощью команды Display

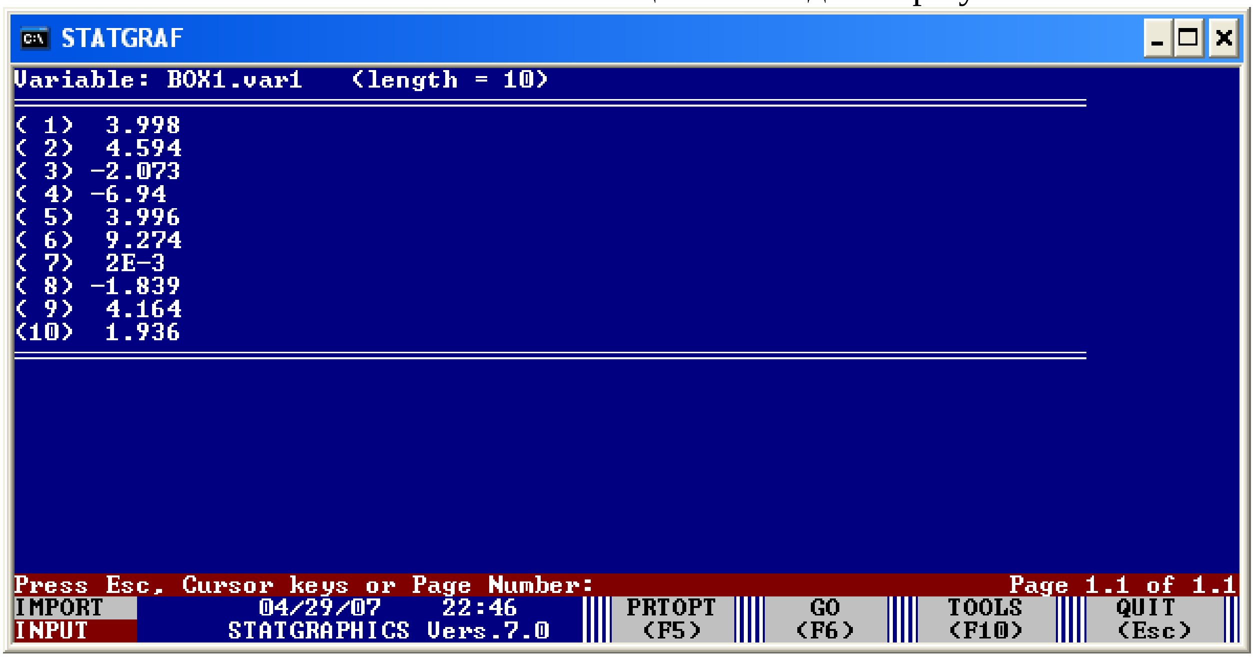

Рис. 3.5. Результат исполнения команды Display для файла box1.var1.

<u>Обработка статистических данных в среде STATGRAPHICS</u>

<u>Команда Descriptive Methods – Summary Statistics, рис.3.6, 3.7, 3.8.</u>

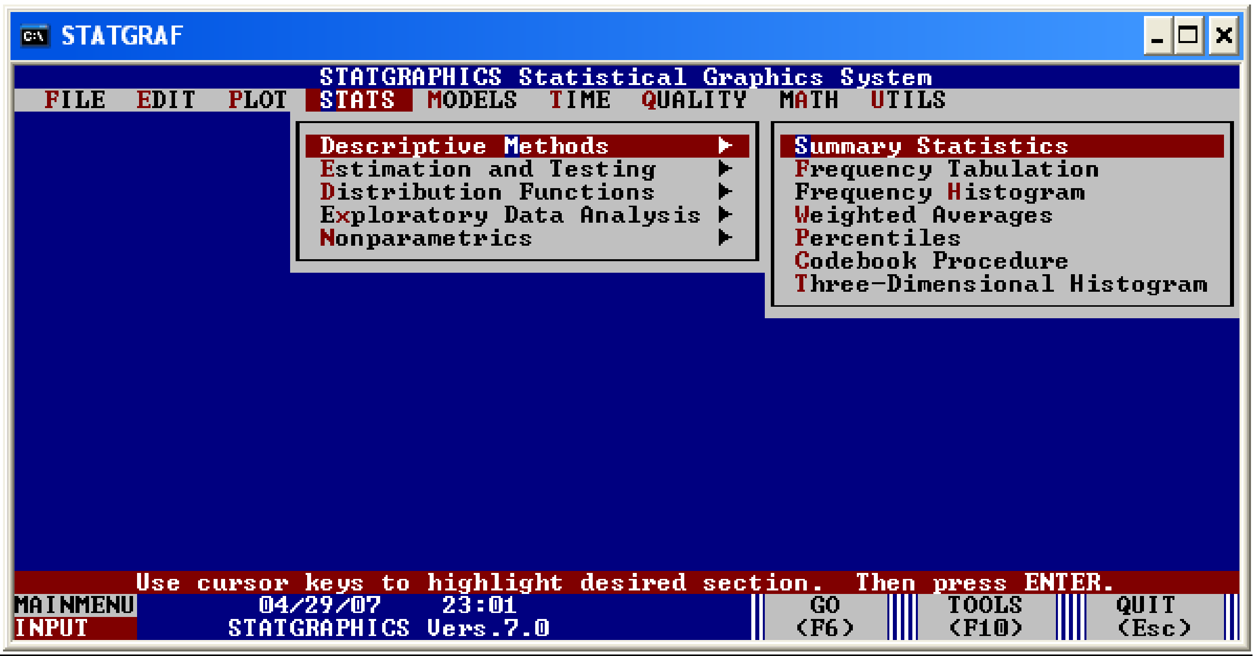

Рис. 3.6. Вызов команды Descriptive Methods – Summary Statistics

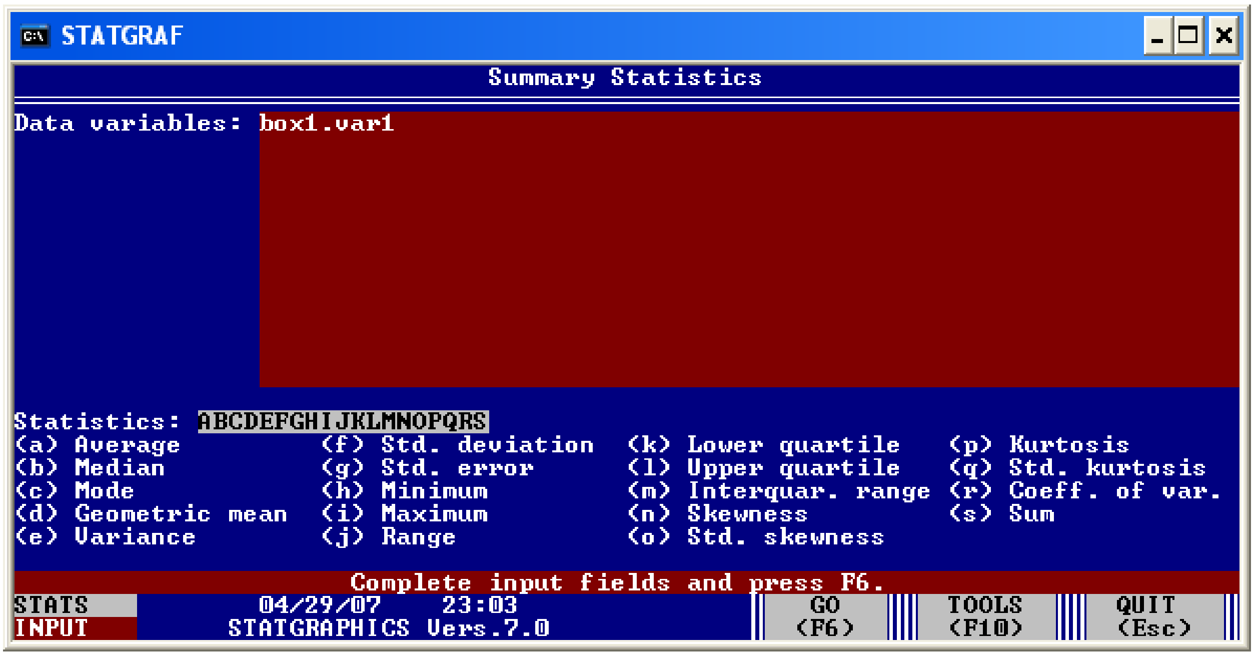

Рис. 3.7. Команда Descriptive Methods – Summary Statistics выполняется для файла box1.var1

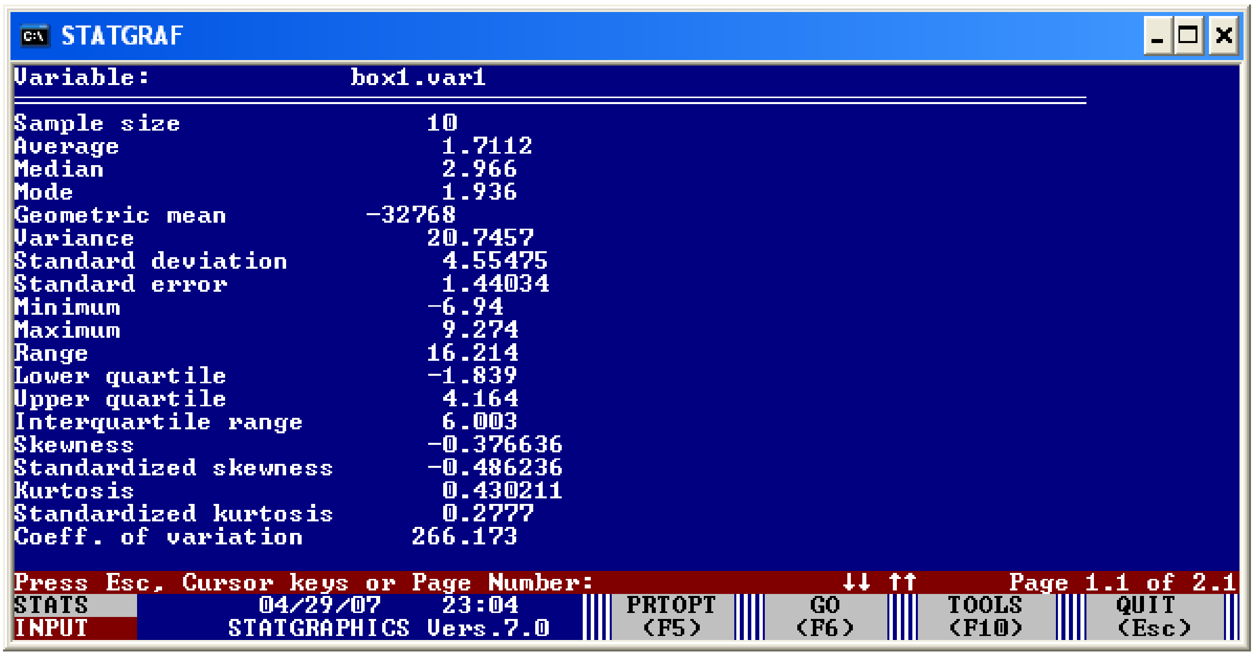

Рис. 3.8. Результаты обработки данных, записанных в файл box1.var1

Проверка гипотезы о соответствии выборочного распределения теоретическому по критерию χ^2 – команда Distribution Functions – Distribution Fitting, рис. 3.9, 3.10, 3.11, 3.12., 3.13.

Рис. 3.9. Вызов команды Distribution Functions – Distribution Fitting

Рис. 3.10. Выбор распределения для сглаживания данных файла box1.var1

Рис. 3.11. Выбор задачи проведения χ^2-теста по данным файла box1.var1

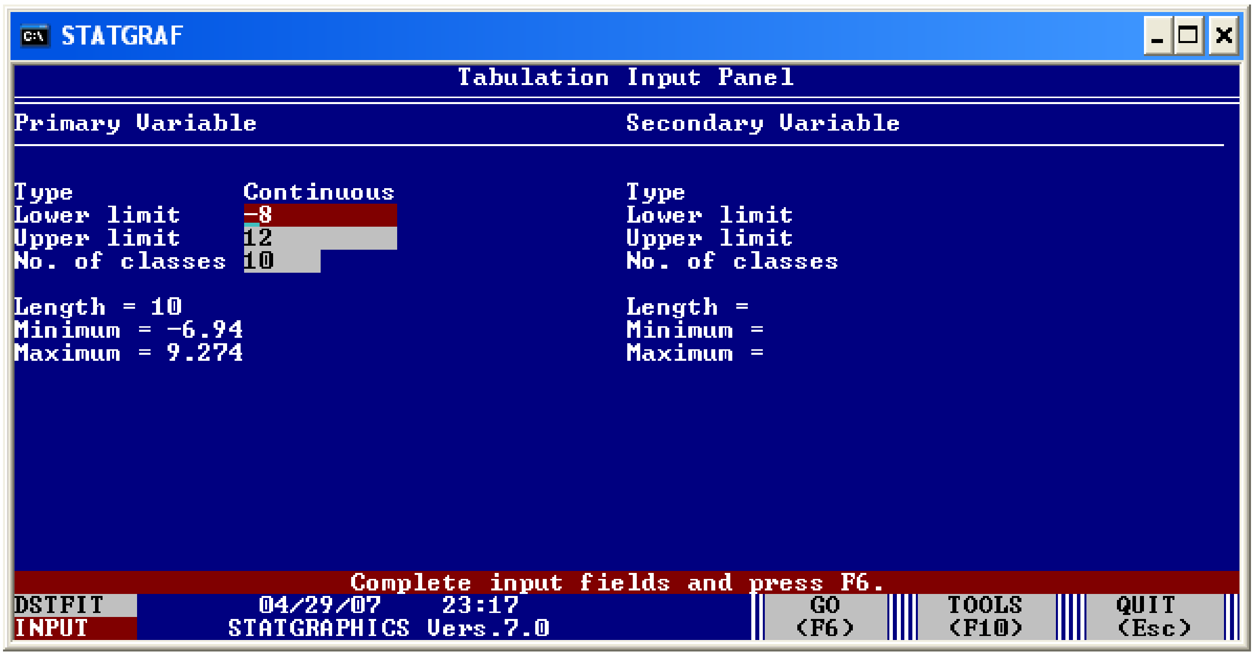

Рис. 3.12. Выбор параметров задачи проведения χ^2 -теста по данным файла box1.var1

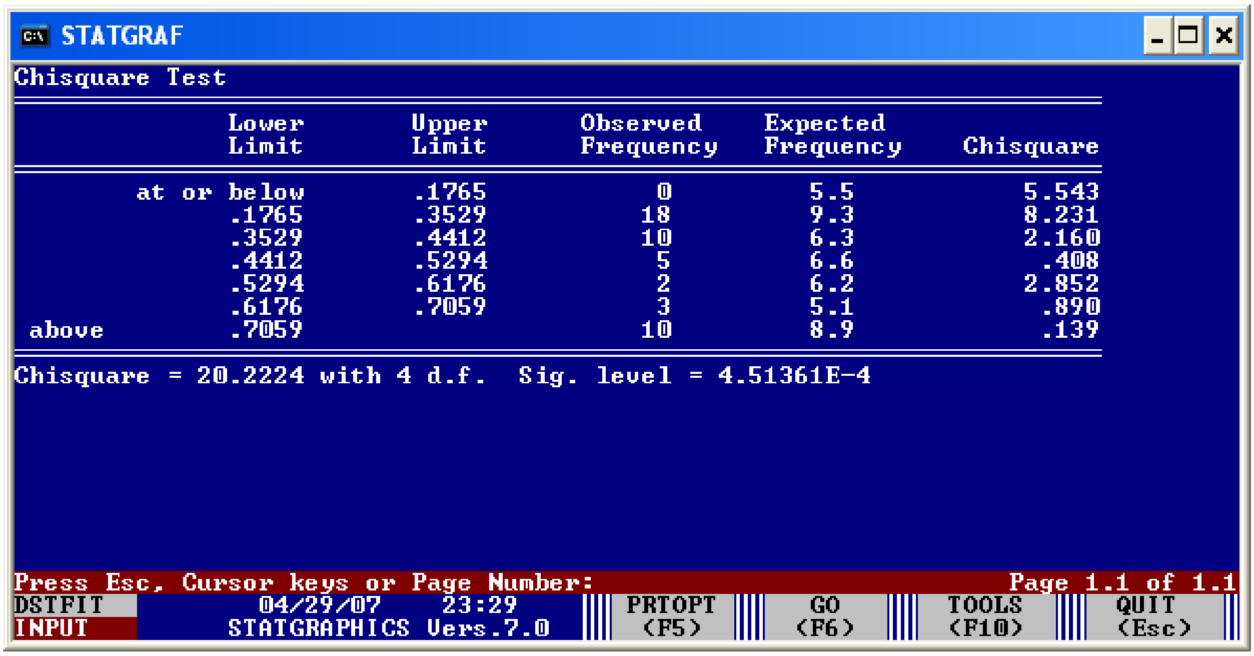

Рис. 3.13. Результат проведения χ^2 -теста по данным файла anova.duration

<u>Формирование набора данных, распределенных по заданному закону с использованием возможностей STATGRAPHICS – команда Distribution Functions – Random Number Generation, рис. 3.14, 3.15, 3.16</u>

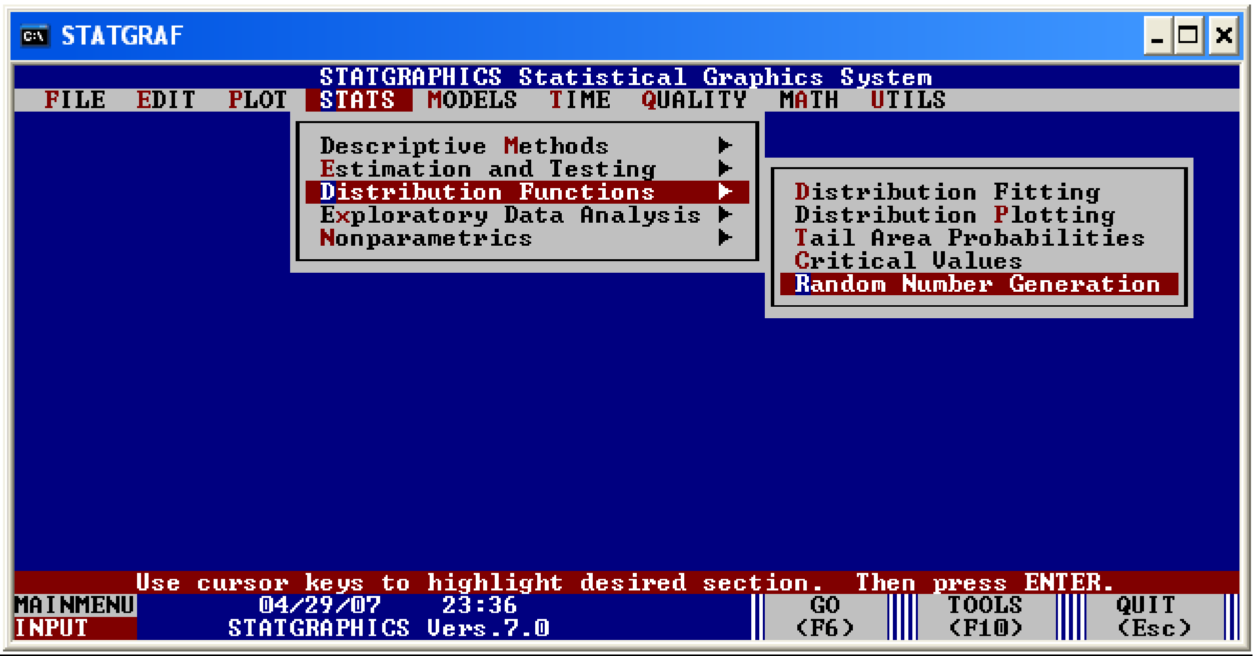

Рис.3.14. Выбор команды Distribution Functions – Random Number Generation

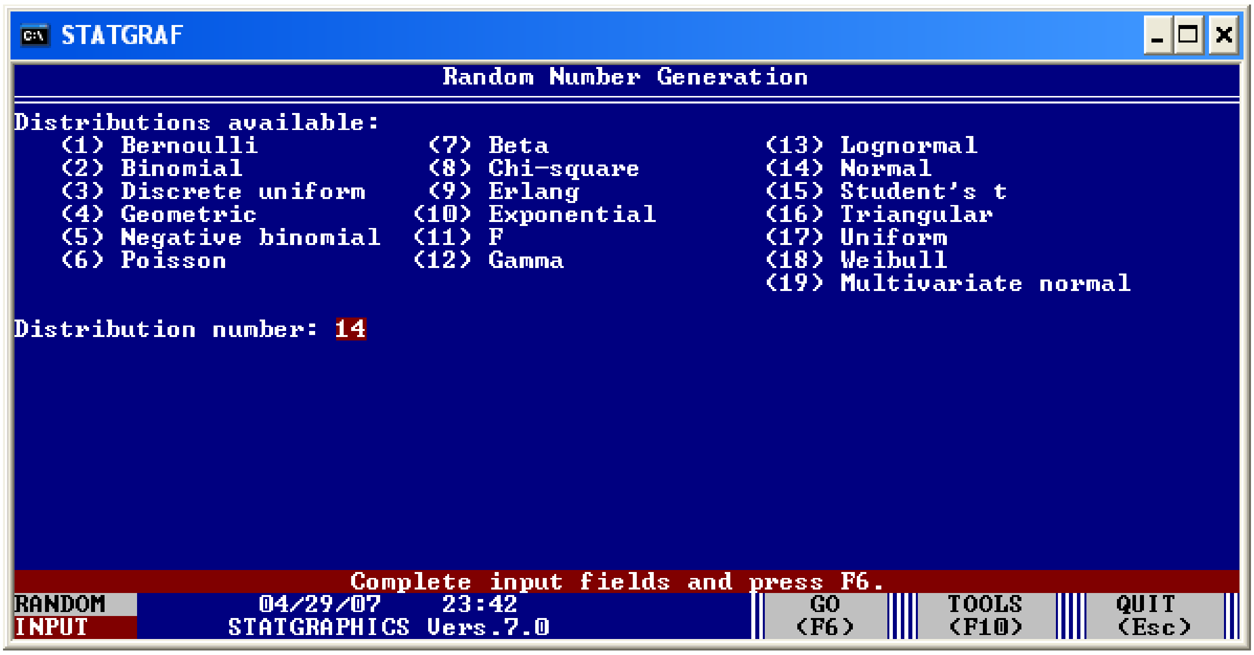

Рис.3.15. Выбор закона генерируемого распределения

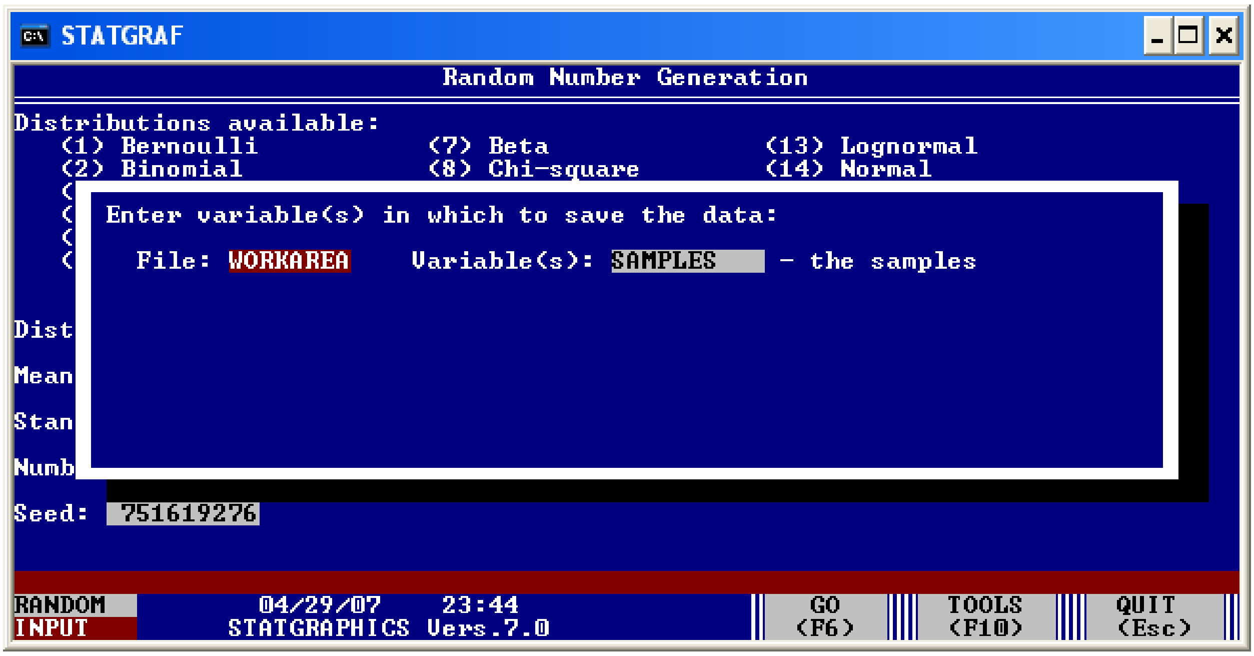

Рис.3.16. Выбор наименования файла и его расширения, например WORKAREA.SAMPLES

3.2. Пакет MATLAB

В зависимости от потребностей пользователя пакет MATLAB может быть инсталлирован с различным числом компонентов, называемых TOOLBOX и представляющих наборы методов, относящихся к соответствующему прикладному разделу математики.

Для целей тестирования программных генераторов может быть использован Statistics Toolbox пакета. Среди многочисленных функций Statistics Toolbox полезными могут оказаться следующие.

1. Построение выборочной (эмпирической) функции распределения – Plot of empirical cumulative distribution function

Syntax cdfplot(x) – обрабатываемые данные должны быть представлены в форме вектора x

h = cdfplot(x) – возвращает график выборочной функции распределения для вектора x.

[h,stats] = cdfplot(x) – помимо графика выборочной функции распределения для вектора x возвращает значения параметров выборки x , поясняемые табл. 2.

Таблица 2

Наименование параметра	Смысловое содержание параметра
stats.min	Минимальное значение
stats.max	Максимальное значение
stats.mean	Среднее арифметическое
stats.median	Медиана
stats.std	Среднее квадратическое отклонение

На рис. 3.17 и 3.18 приводятся формат команды **cdfplot** и результаты обработки данных, представленных файлом box1.txt.

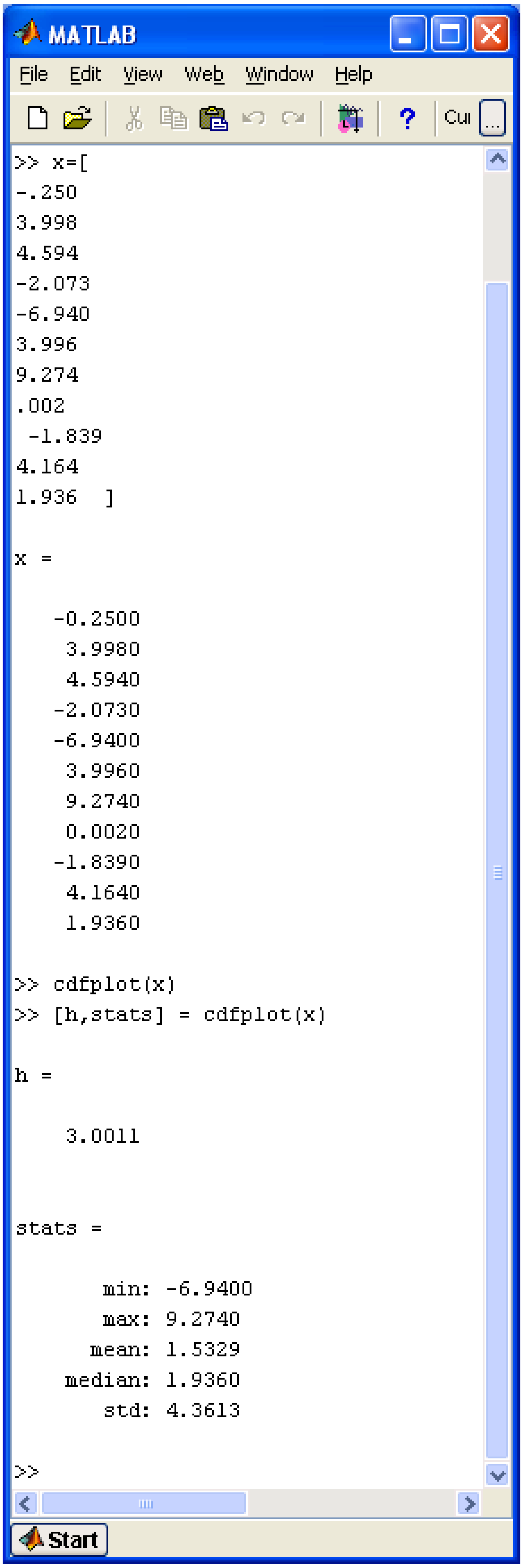

Рис. 3.17

Вектор x формируется из текстового файла box1.txt. в командном окне MATLAB, затем выполняется команда **[h,stats] = cdfplot(x)**. На рис. 17 приводится график выборочной функции распределения и функции стандартного нормального распределения. Оба графика получены посредством выполнения следующей последовательности команд MATLAB:

```
xx = -7:.1:10;
cdfplot(x)
hold on
plot(xx,normcdf(xx),'r--')
```

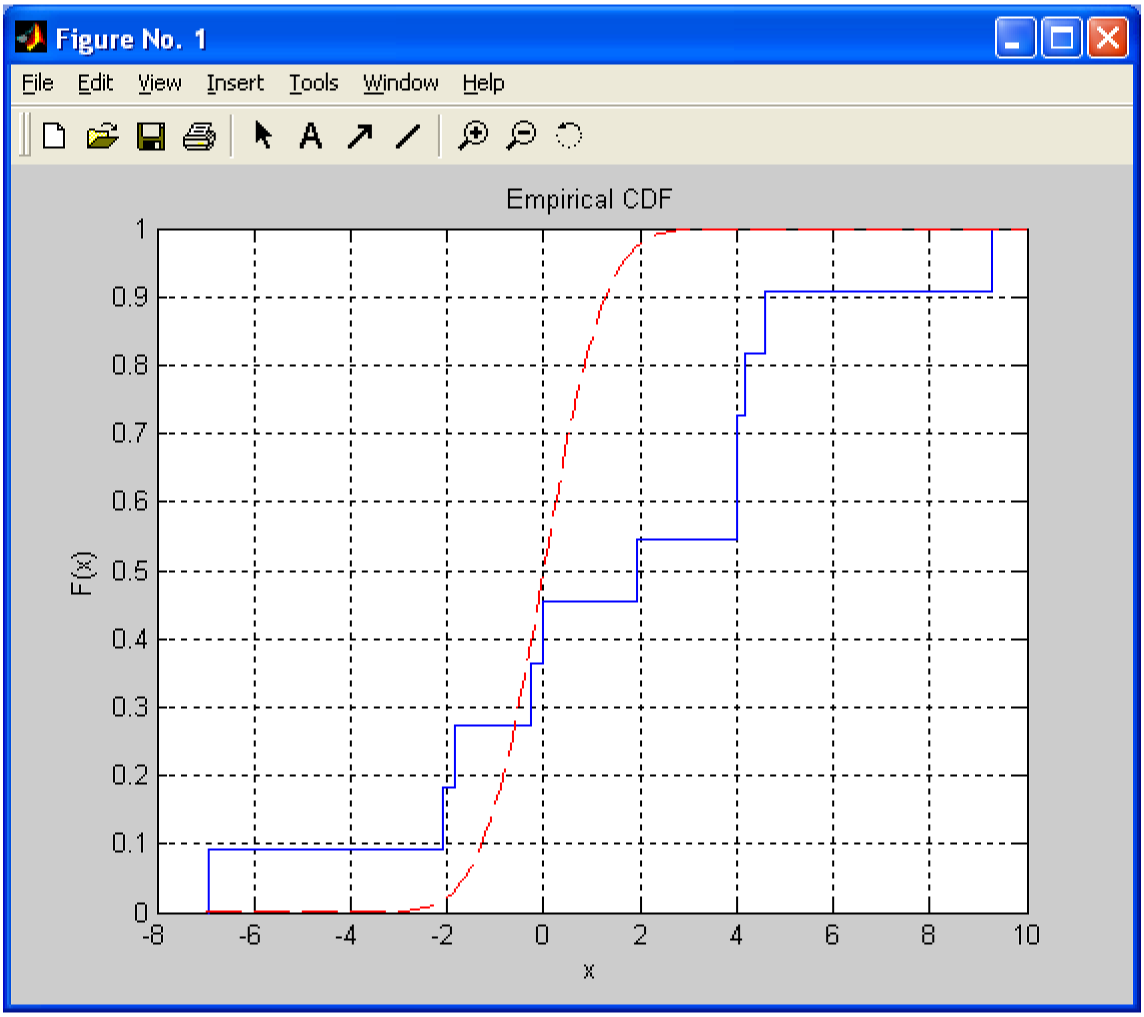

Рис. 3.18

2. **Тест Колмогорова-Смирнова для проверки гипотезы о законе распределения – kstest Kolmogorov-Smirnov test of the distribution of one sample**

Syntax

H = kstest(x)

H = kstest(x) – обеспечивает тестирование данных вектора x по критерию Колмогорова-Смирнова на предмет соответствия стандартному нормальному распределению $N(0,1)$ с нулевым мат. ожиданием и единичной дисперсией. Процедура возвращает 0, если принимается нулевая гипотеза (с 5-ти % уровнем значимости) о соответствии выборочного распределения $N(0,1)$ и 1 в противном случае.

H = kstest(X,cdf)

H = kstest(X,cdf) – обеспечивает сравнение выборочного распределения с распределением, задаваемым таблицей cdf с двумя столбцами, первый из которых представляет значения случайной величины (вектора x), а второй – соответствующие значения функции вероятностей. Если какое-либо значение вектора x окажется не представленным в таблице cdf, соответствующая строка таблицы будет получена посредством линейной интерполяции. Важно, чтобы все значения вектора x находились в интервале от минимального до максимального чисел первого столбца таблицы cdf.

H = kstest(x,cdf,alpha,[])

H = kstest(x,cdf,alpha,[]) – обеспечивает возможность задания произвольного уровня значимости при проверке нулевой гипотезы

Тестирование данных вектора x (порожденного файлом box1.txt) с помощью команды **H = kstest(x)** привело к непринятию нулевой гипотезы о соответствии выборочного распределения стандартному нормальному (рис. 3.19), что подтверждается сравнением графиков функций распределений на рис. 3.18.

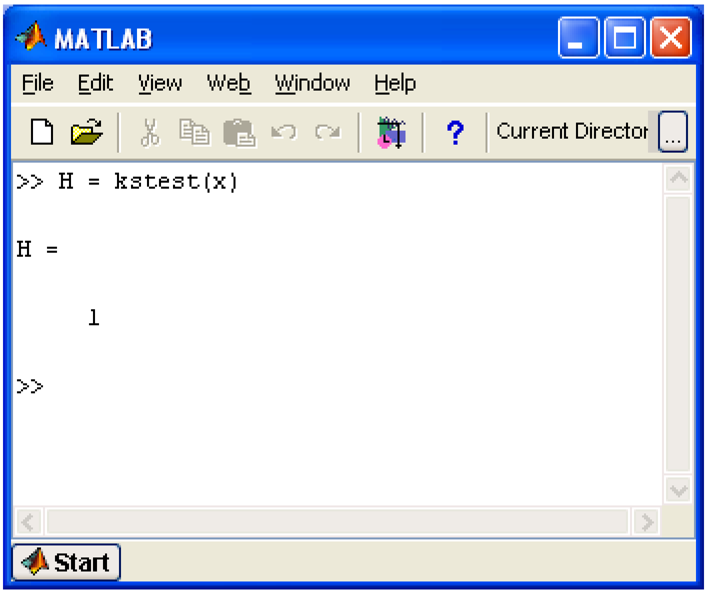

Рис. 3.19

Литература

1. Колесников Д.Н., Сиднев А.Г., Юрганов А.А. Моделирование случайных факторов в задачах автоматики и вычислительной техники. Учебное пособие. Санкт Петербург. – Л.: СПбГТУ. 1994.– 104 с.
2. Быков В.В. Цифровое управление в статистической радиотехнике. – М.: Советское радио. 1971. – 324 с.
3. Бусленко Н.П. Моделирование сложных систем. – М. Наука. 1977. – 204 с.