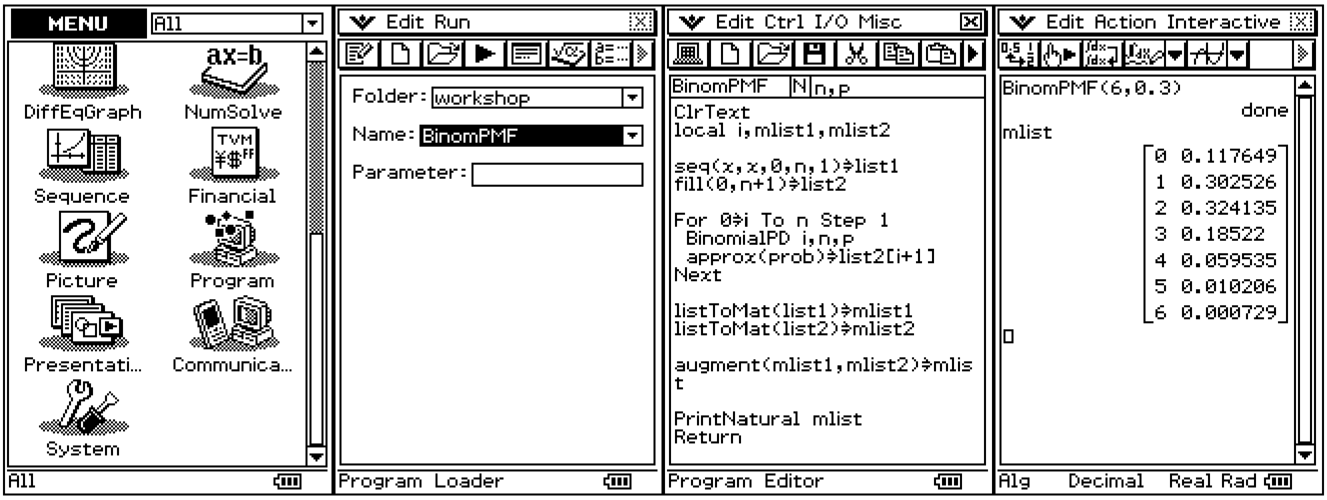
worksheet

Using the ClassPad300Plus in Statistics to Draw Step Functions and to Compute their Quantiles (Workshop)

Ludwig Paditz, University of Applied Sciences Dresden (FH), Germany

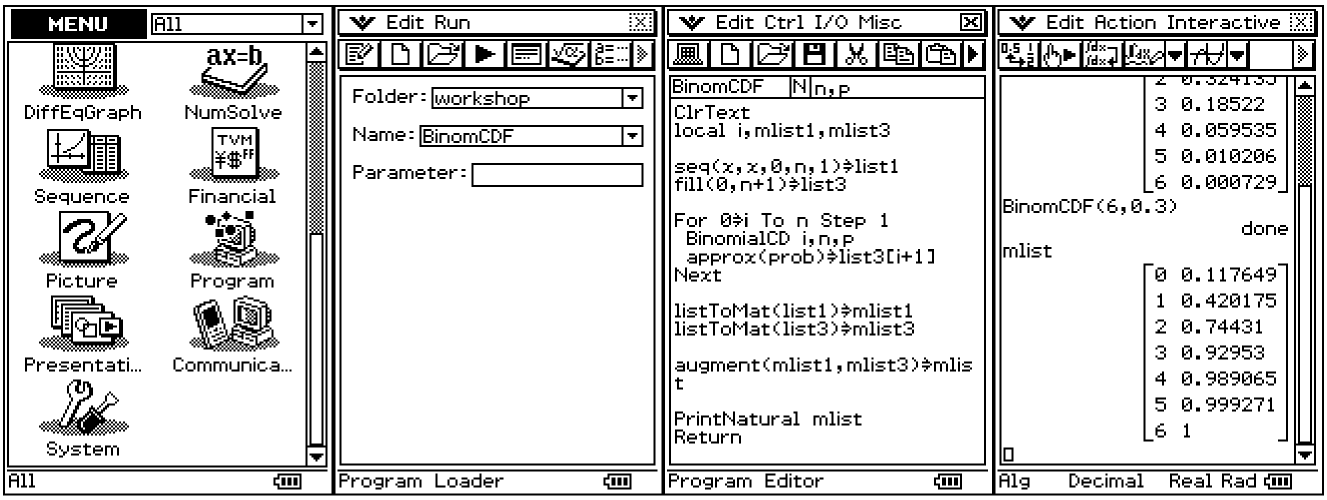
Let us consider the ClassPad300Plus (with the new operating system OS 03.02) and discuss on some new exercises in statistics, e.g. drawing step-functions and computing their quantiles.
We consider a ***B*(*n*,*p*)**-distribution function (binomial distribution with the parameters ***n*** = 6 and ***p*** = 0.3 say).

1st step: have a look in the program **BinomPMF(n,p)** to compute the values of the single probabilities (PMF … probability mass function) and compute the mlist:

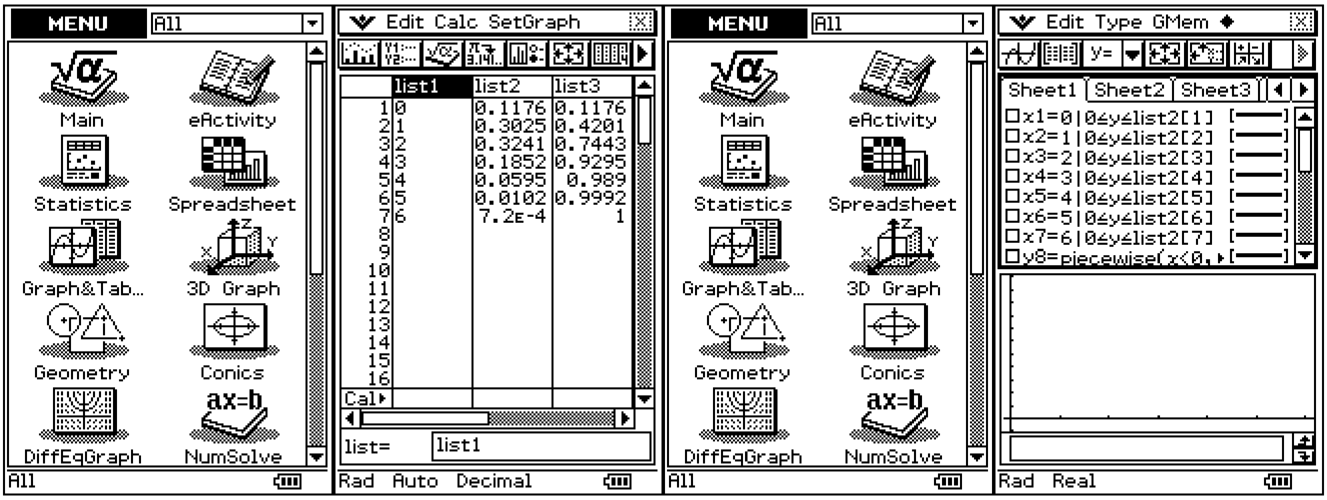


In the program the ClassPad command BinomialPD is used. The result (a table or better the matrix mlist) we can see in the main menu.

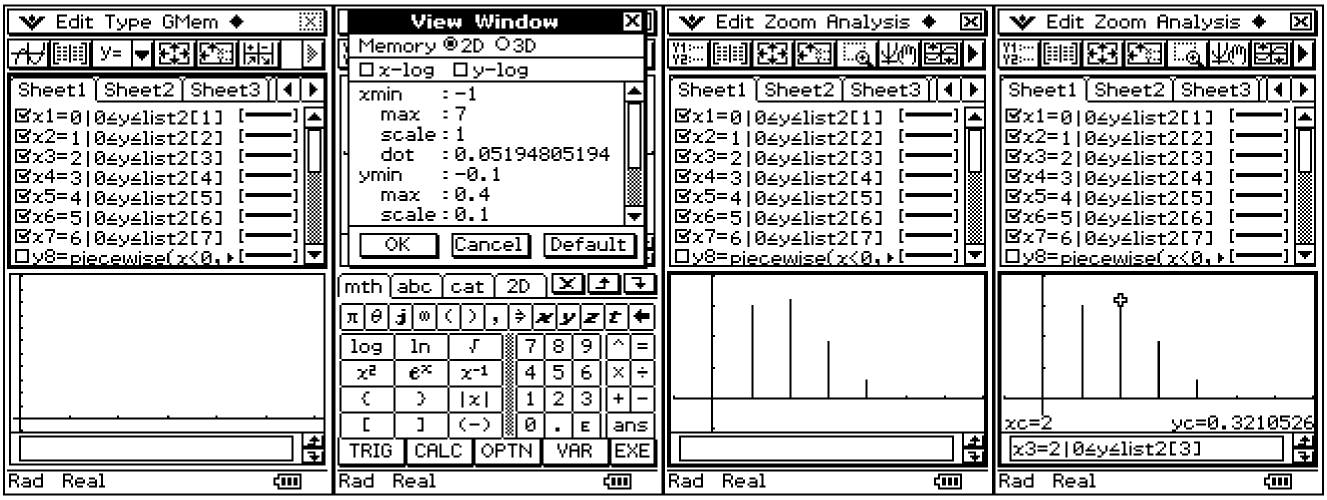
2nd step: have a look in the program **BinomCDF(n,p)** to compute the values of the cumulated probabilities (CDF … cumulative distribution function) and compute the mlist:

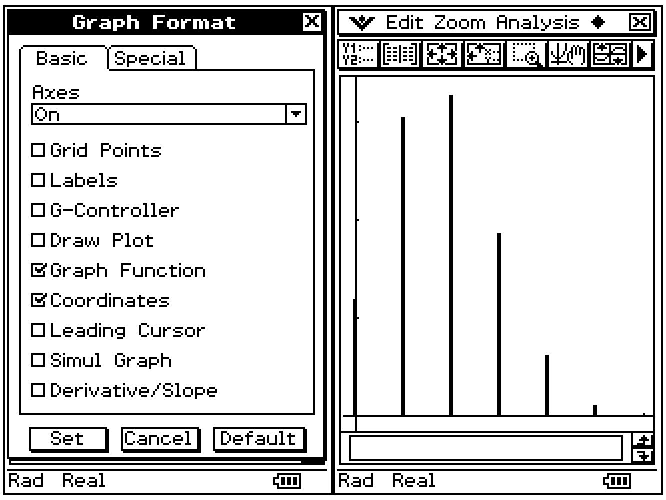


3rd step: Let us draw the **line diagram** for the probability mass function **P(X=x)** by the help of CP300. We use our small program BinomPMF and suppose **p=0.3** and **n=6**.
In the 1st step we have already computed the list1 and list2 respectively (contained in mlist). Now we open the **Statistics menu** (stat editor) to have a look in the list1 and list2. Than we change to the graph editor (**Graph&Table menu**) and input the terms for the line diagram. Every line is a singe x-function.



We need the functions x1 up to x7 and activate these functions. Than we draw the wished graphics. Don't forget to input a good **view window** and finally try to trace the line diagram.
Study **the syntax** of the x-functions: e.g. x2=1|0≤y≤list2[2] and switch off "**Draw Plot**"!

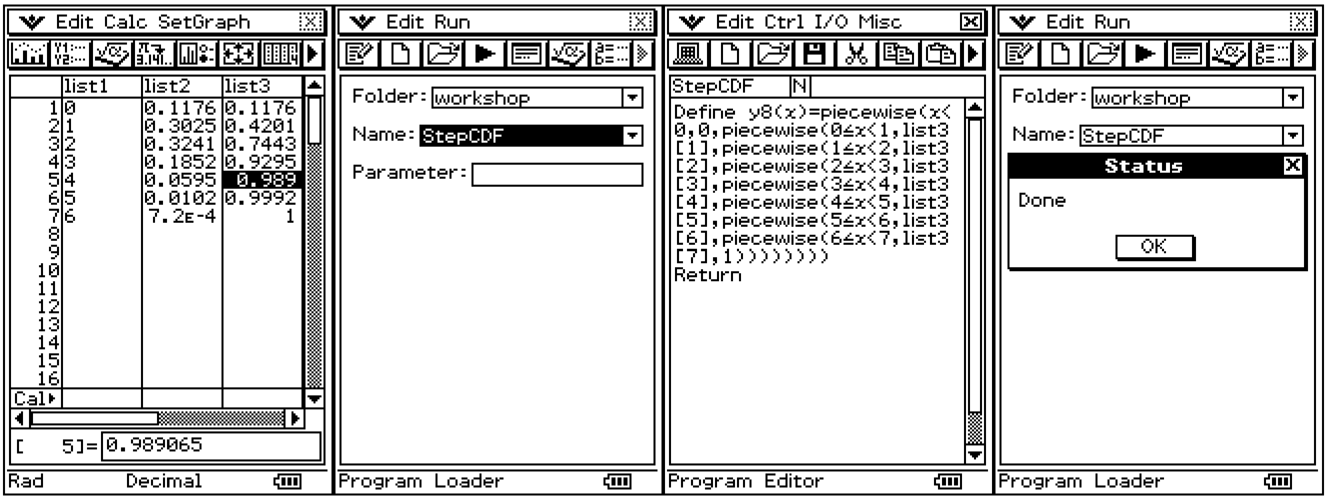




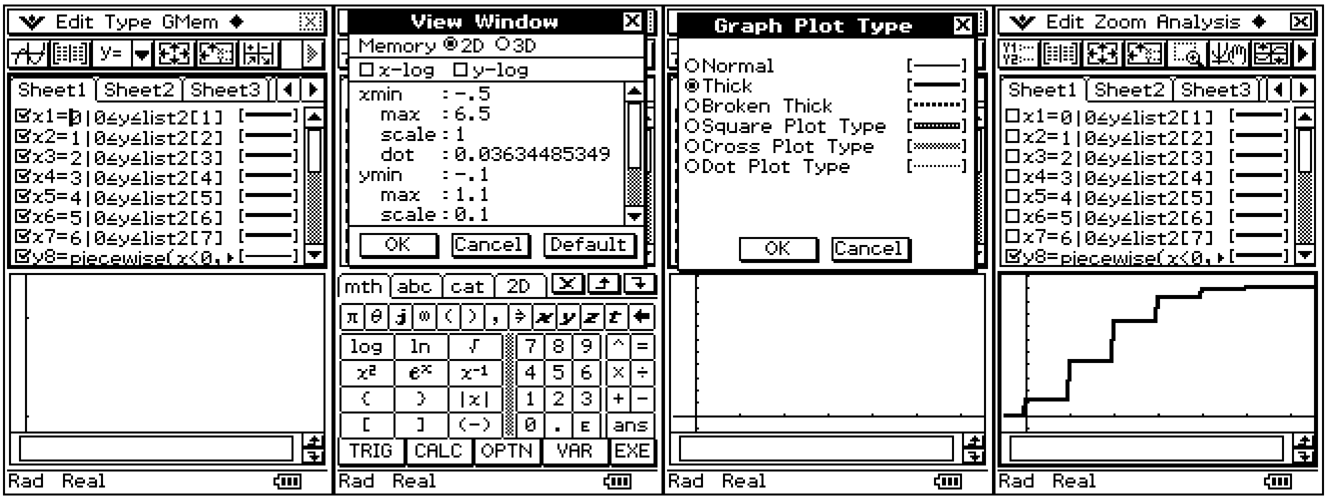
with Resize and Zoom Box

4th step: Let us draw the step function for the cumulative distribution function by the help of CP300. We already used the program BinomCDF to generate a table of values and suppose **p=0.3** and **n=6**. In the 2st step we have computed the list1 and list3 respectively (contained in mlist). Now we open the **Statistics menu** (stat editor) to have a look in the list1 and list3.

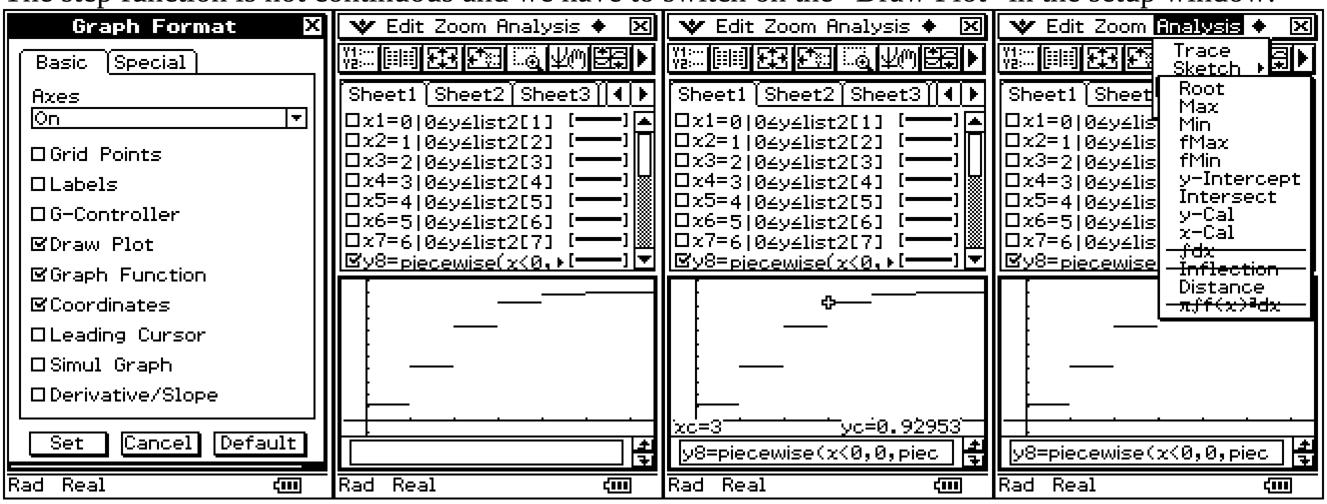
The screenshots are from Stat Editor and Program Editor and Program Loader respectively. Now we can define the step function by the help of the **piecewise function** and the program **StepCDF** and we start StepCDF in the Program Loader:



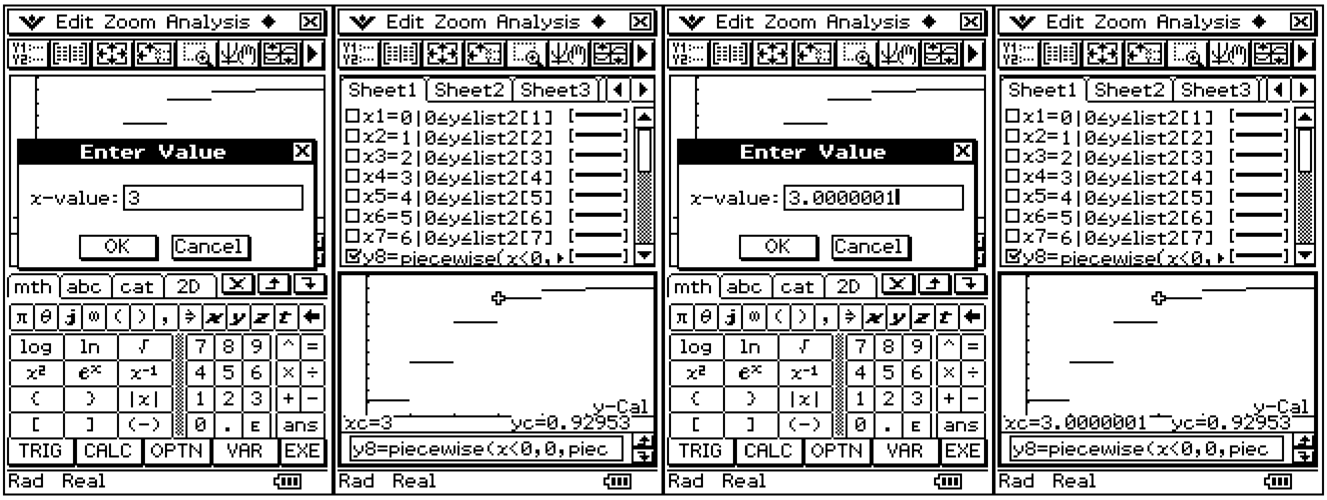
Now have a look in the graph editor and activate only y8 and draw the step function:

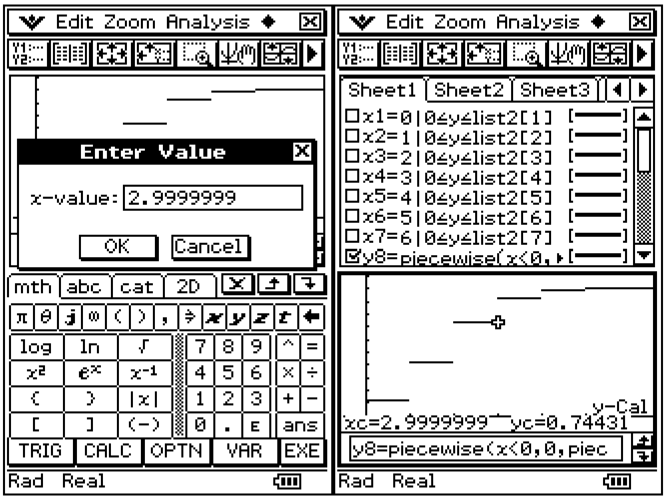


The step function is not continuous and we have to switch on the "Draw Plot" in the setup window:



Now try **Trace** and **G-Solve** and compute in the graph menu values of the step function. Input x=3 and x=3.0000001 and x=2.9999999 to check the step function is right continuous:

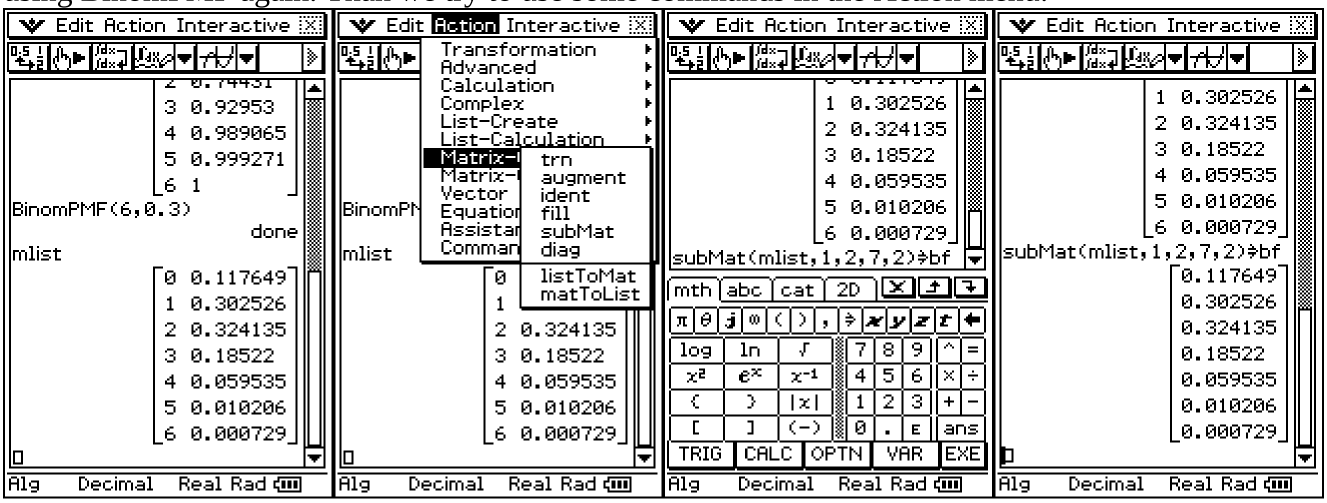




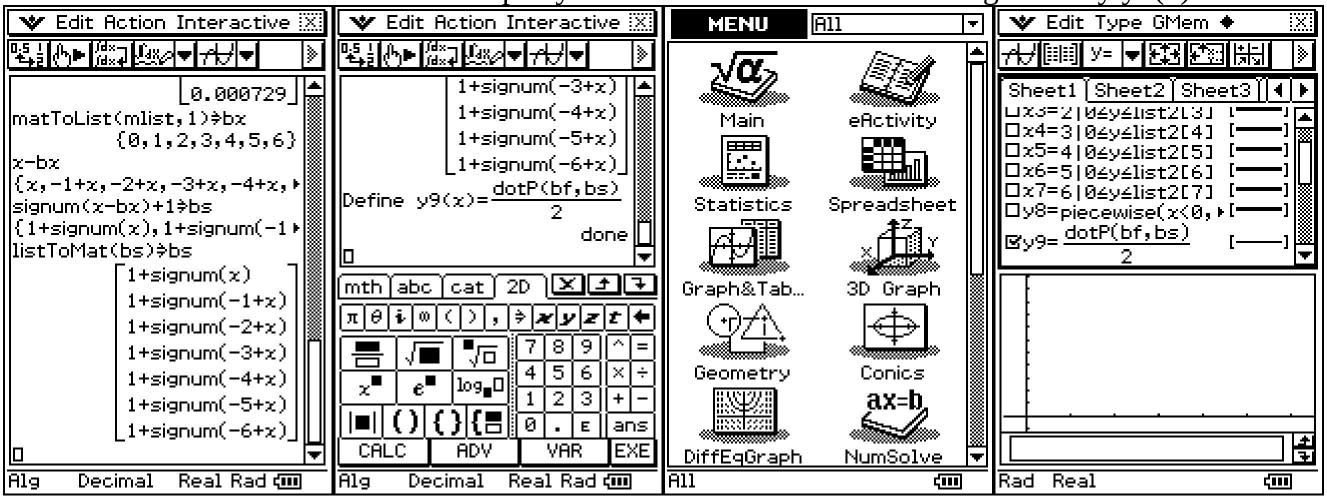
5th step:

Now we define a step function by the help of the **vector calculation** to see another possibility to get step functions. However this definition shows: we have no value in the discontinuous points, thus the piecewise definition is a better one.

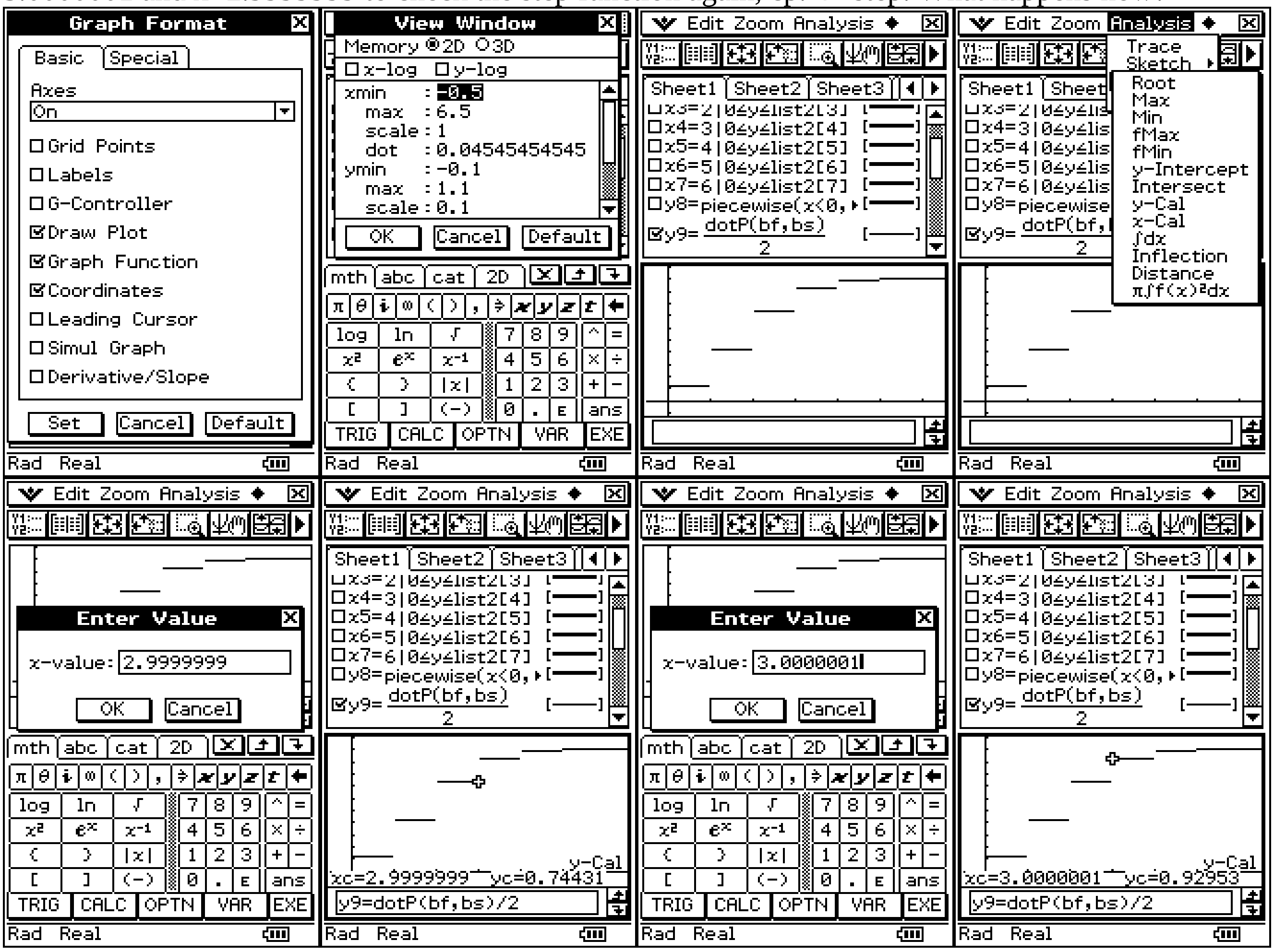
We try to draw the step function ***F*(*x*) = *P*(*X* ≤ *x*)**, where ***X*** is a ***B*(*n*,*p*)**-distributed random variable. By the help of the **signum**-function we can generate the wished step function, using the possibilities of the algebra. We work now in the **main** menu. We start with the single probability mass function values using BinomPMF again. Than we try to use some commands in the Action menu:

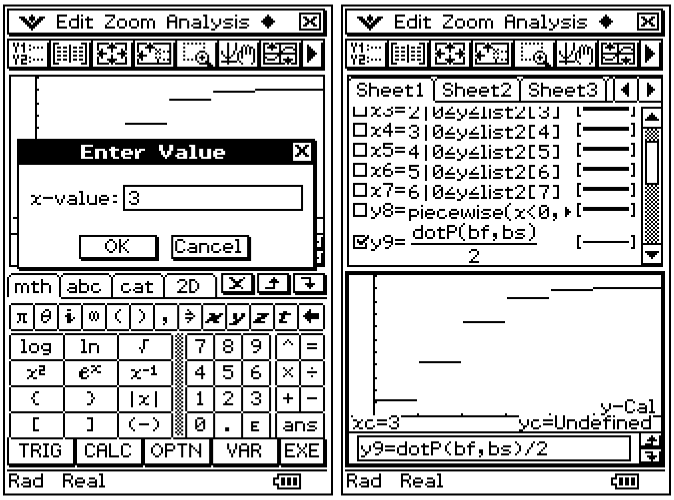


Explain the syntax of the **subMat(mlist,1,2,7,2)** function! Than we use the matToList and ListToMat functions to convert parts of a matrix into a list or to convert a list into a matrix (vector).
We shift the discontinuous point of the signum function (discontinuous point is 0) into the probability mass points 0, 1, 2, …, n of the binomial distribution by the help of signum(x-bx). We can apply the signum function with a list-valued argument x-bx !
The ClassPad can interpret the difference x-bx ! Explain what is x-bx !
signum(x-bx)+1 creates a list of x-terms for our step function! Than we transform this list into a vector to use vector calculation in the next step. By means of the **dotP** function we get finally y9(x).



Now open the Graph editor: you can discover y9(x) already in the editor, created in the Main menu! We graph the step function. Don't forget to switch on "Draw Plot" in the Graph Format menu. Than try **Trace** and **G-Solve** and compute in the graph menu values of the step function. Input x=3 and x = 3.0000001 and x=2.9999999 to check the step function again, cp. 4th step. What happens now?





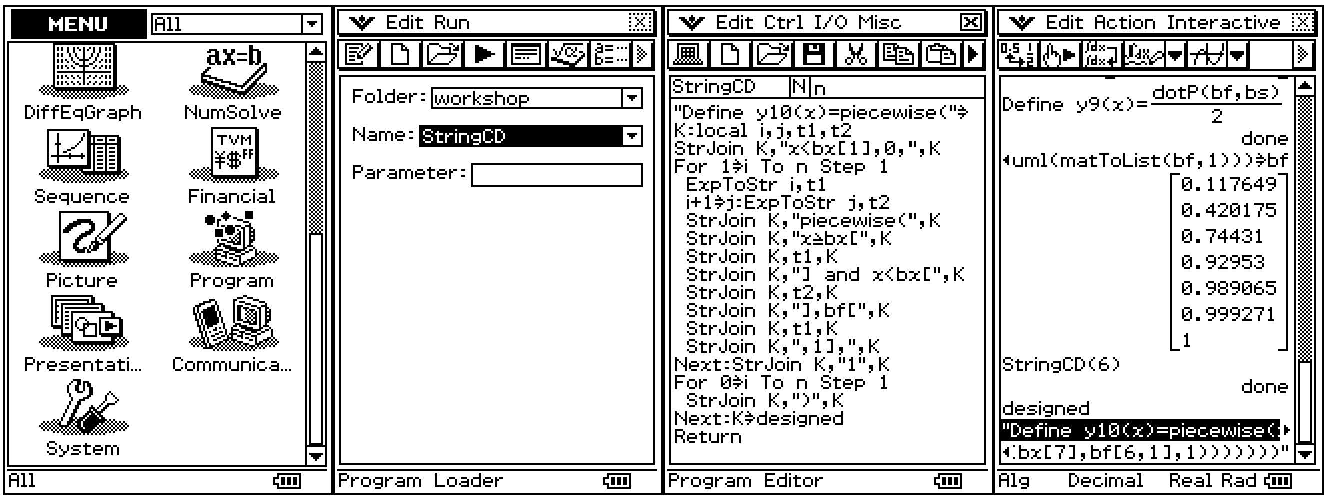
y9(3) = undefined!

This definition shows that we have no value in the discontinuous points, thus the piecewise definition is a better one to define step functions in the probability theory.

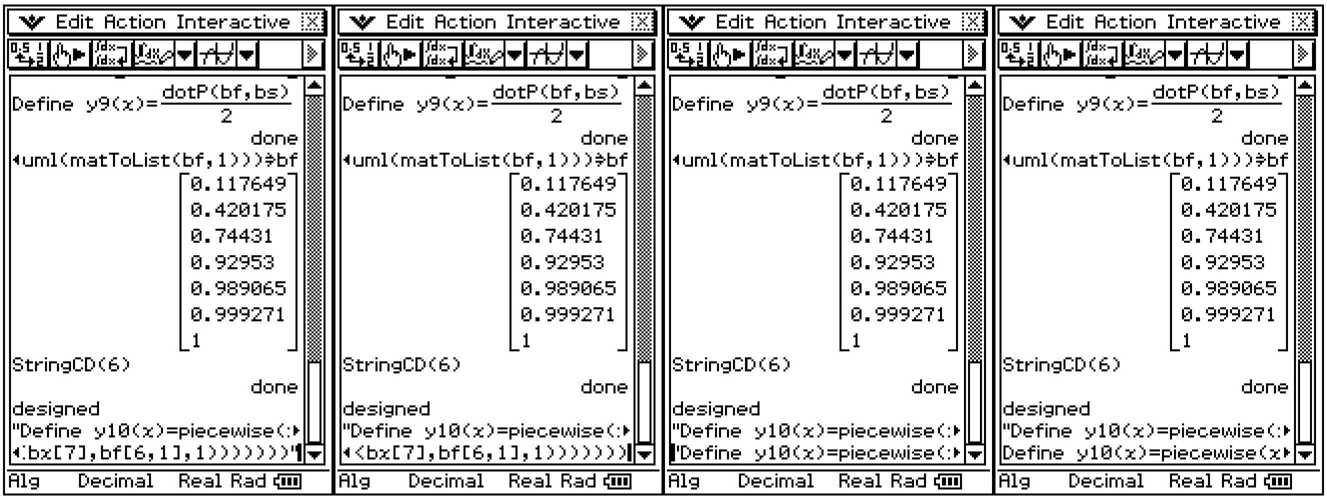
6th step:
It is possible to create a long iterated term of a function by the help of string commands.
Again using the **piecewise**-function, we can generate and draw an exact right continuous step function:

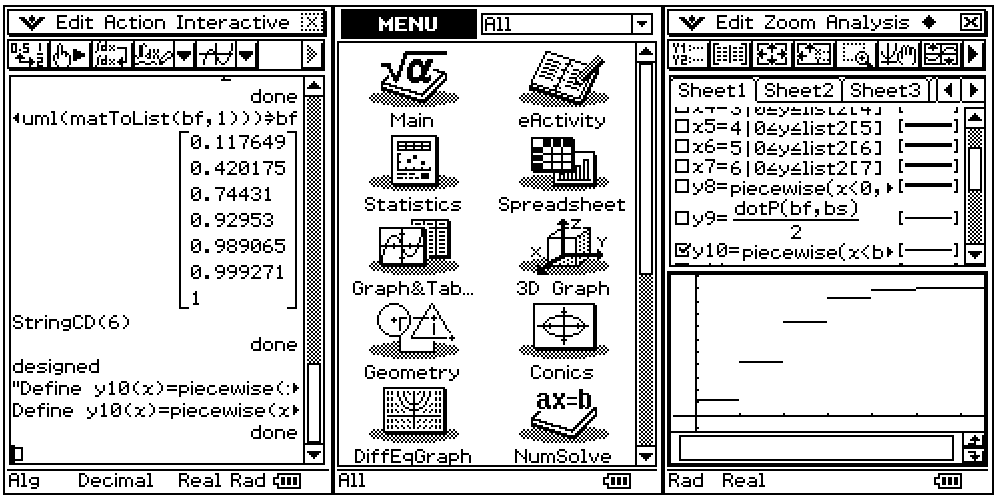
y8(x) =piecewise(x<bx[1],0,
piecewise(x≥bx[1] and x<bx[2],bf[1,1],
piecewise(x≥bx[2] and x<bx[3],bf[2,1],
piecewise(x≥bx[3] and x<bx[4],bf[3,1],
piecewise(x≥bx[4] and x<bx[5],bf[4,1],
piecewise(x≥bx[5] and x<bx[6],bf[5,1],
piecewise(x≥bx[6] and x<bx[7],bf[6,1],1)))))))

Here the list **bx** ={0,1,2,3,…,n} and the vector **bf** = $[p_0, p_1, p_2, \ldots, p_n]^T$ are used. We generate this large term for y8(x) by the help of the program **StringCD** using commands for strings and characters now.

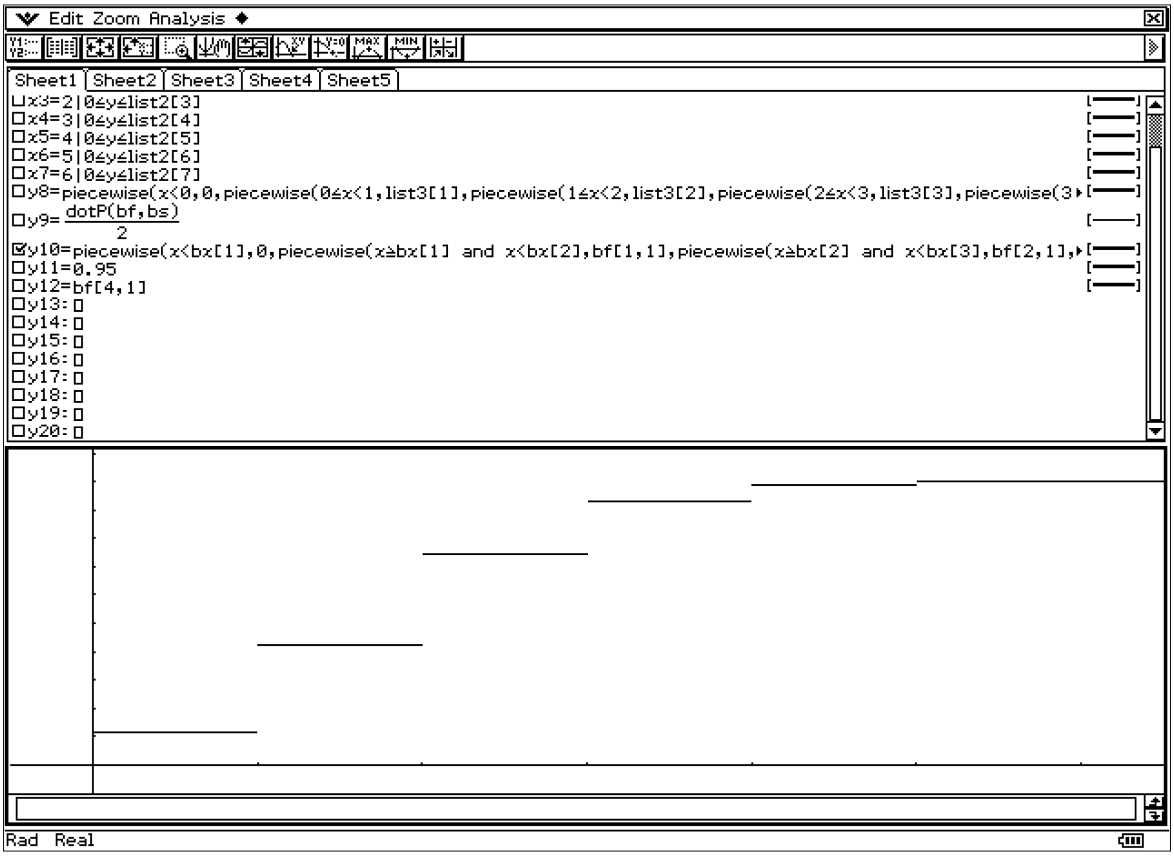


At first have a look in the Program editor with **StringCD**, than start this program in Main menu and call the result **designed**. Be sure that the cumulated probabilities are in bf now! Use at first the **listToMat(cuml(matToList(bf)))** syntax to transform the single probabilities in the vector bf with the cumulated probabilities! Input the answer of **designed** in a new line and delete the strings “ ”.
Than you can realize the **Define** command without problems to define y10(x)!





Have a look in the Graph editor to see that the y8 and y10 functions are the same functions!



7^{th} step:

Computing of the quantiles of the binomial distribution (graphic solution)

Finally we study possibilities to compute the quantiles for a given probability ***prob***, i.e. we try to solve the inequality $\boldsymbol{F(x\text{-}0) = P(X < x)} < \boldsymbol{prob} \leq \boldsymbol{F(x) = P(X \leq x)}$ with an appropriate $\boldsymbol{x}$. Here we can use the **InvBinomialCD**-function of the ClassPad300.

Remember: We use the following definition of the quantile of order *gamma*:

The number $\mathbf{b}_{gamma}$, $\mathbf{0 < \mathit{gamma} < 1}$, of a **B(n,p)**-distributed random variable **X** is called **quantile** of the order ***gamma***, if the inequality $\mathbf{P(\ X < b_{\mathit{gamma}}\ ) \leq \mathit{gamma} \leq P(\ X \leq b_{\mathit{gamma}}\ )}$ is fulfilled.

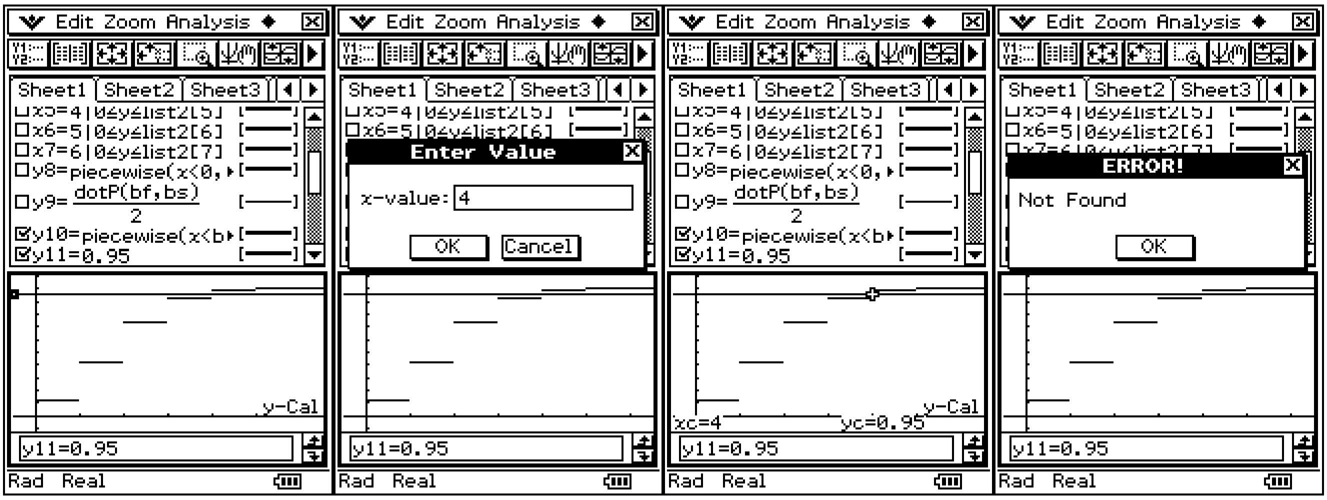
The **B(n,p)**-distributed random variable **X** has a discrete distribution function **F(x)** (a right continuous step-function).
If the equation ***gamma* = F(x)** has no solution, than the wished quantile $\mathbf{b}_{gamma}$ is the integer number **x**, which fulfils the inequality $\mathbf{F(x\text{-}0) < \mathit{gamma} < F(x) = F(x+0)}$.

In the other case (the equation ***gamma* = F(x)** has (at least) one solution) the number $\mathbf{b}_{gamma}$ is not determined in an unique manner and the wished quantile is one of the solutions of the equation ***gamma* = F(x)** with a value **x** of the interval **[k, k+1)**, in which the (cumulative) distribution function **F(x)** has the value **F(x) = *gamma***, or we choose **x=k+1**.

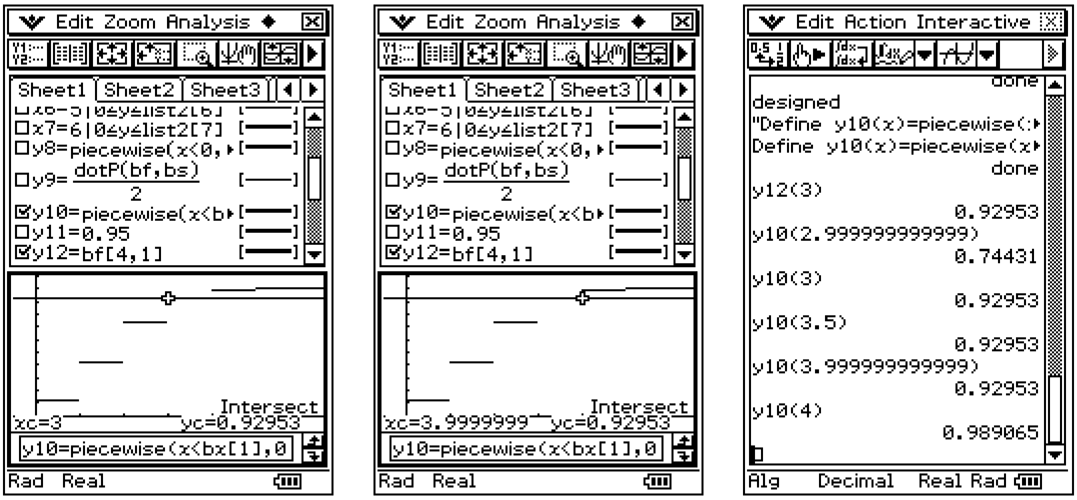
In the statistical theory the quantiles are called critical values and we need these values to construct confidence intervals or to study statistical hypotheses. In such cases the probability ***gamma*** or **1-*gamma* = *alpha*/2** is called confidence level or significance level.

The binomial distribution has the well-known probability density function, given by $\mathbf{P(X=x) = C(n,x) * p^{n} * (1\text{-}p)^{n\text{-}x}}$ for **x = 0, 1, 2, …, n.** Here **C(n,x) = n!/(x!(n - x)!)** is the binomial coefficient. Sometimes the probability density function is called probability mass function.
Note that this function is the general term in the binomial expansion of $\mathbf{(p + (1\text{-}p))^{n}}$.

Start: Computing of the quantiles of the binomial distribution (graphic solution)
Again let be **n=6** and **p=0.3** and ***gamma*=0.95.** We try to solve the equation **F(x)=0.95** by the help of the intersection of **y10(x)=F(x)** and **y11(x)=0.95** and get no solution (cp. ERROR).

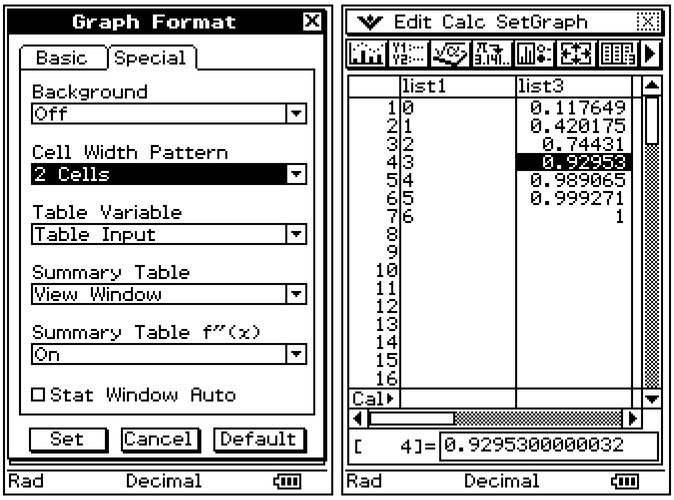


In the graphic representation we can observe, the solution is $\mathbf{x = b_{\mathit{gamma}} = 4}$, because this value fulfils the definition $\mathbf{F(x\text{-}0) = P(\ X < b_{\mathit{gamma}}\ ) \leq \mathit{gamma} \leq P(\ X \leq b_{\mathit{gamma}}\ ) = F(x+0) = F(x).}$
Now let be ***gamma* = 0.92953** [= F(3)]. Put **y12(x) = bf[4,1] = 0.92953** and see what happens with the intersection. We get the smallest graphical solution $\mathbf{x = b_{\mathit{gamma}} = 3.}$



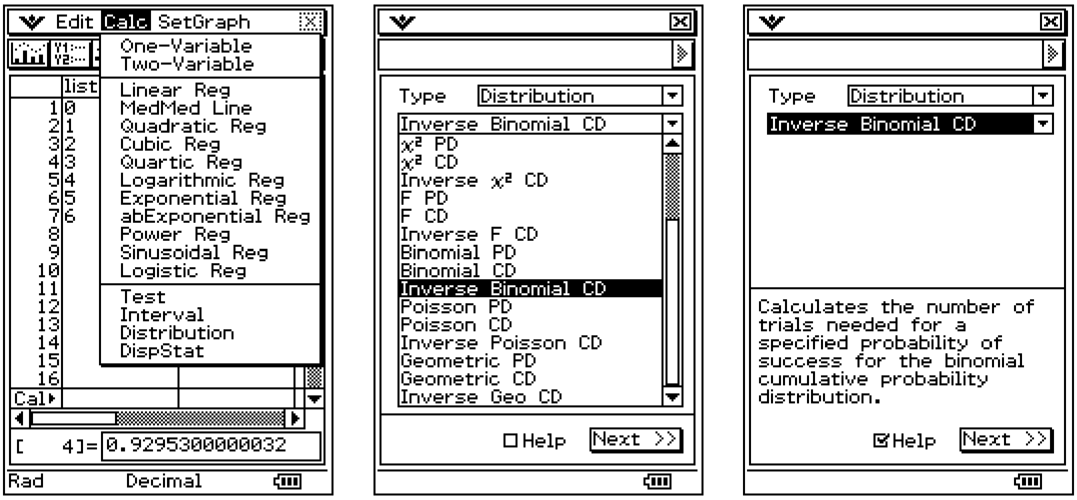
Finally **3 ≤ b**$_{gamma}$ **≤ 4** is the set of all solutions of **P(X < b**$_{gamma}$ **) ≤ *gamma* ≤ P(X ≤ b**$_{gamma}$ **)**.

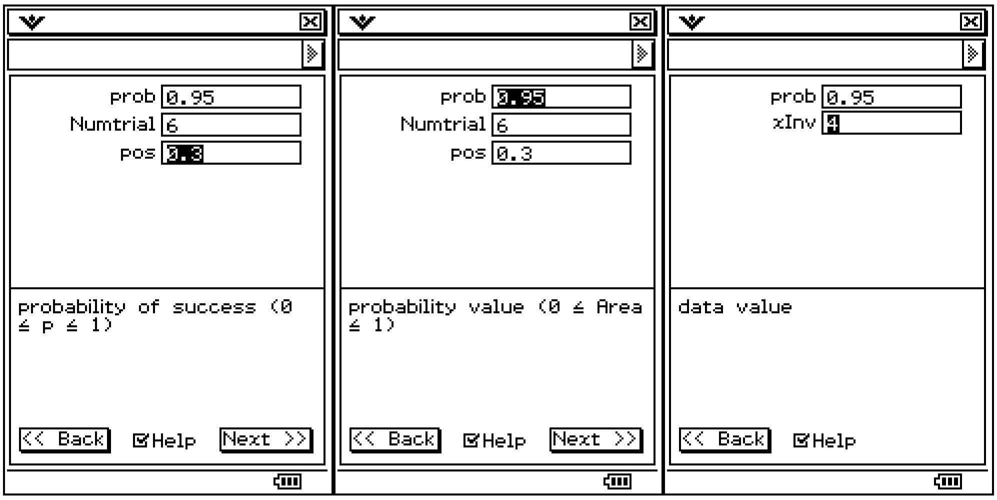
The same solutions we get considering the table of the values of the function **F(x)**, cp. list1 and list3 in the Stat-List-Editor.

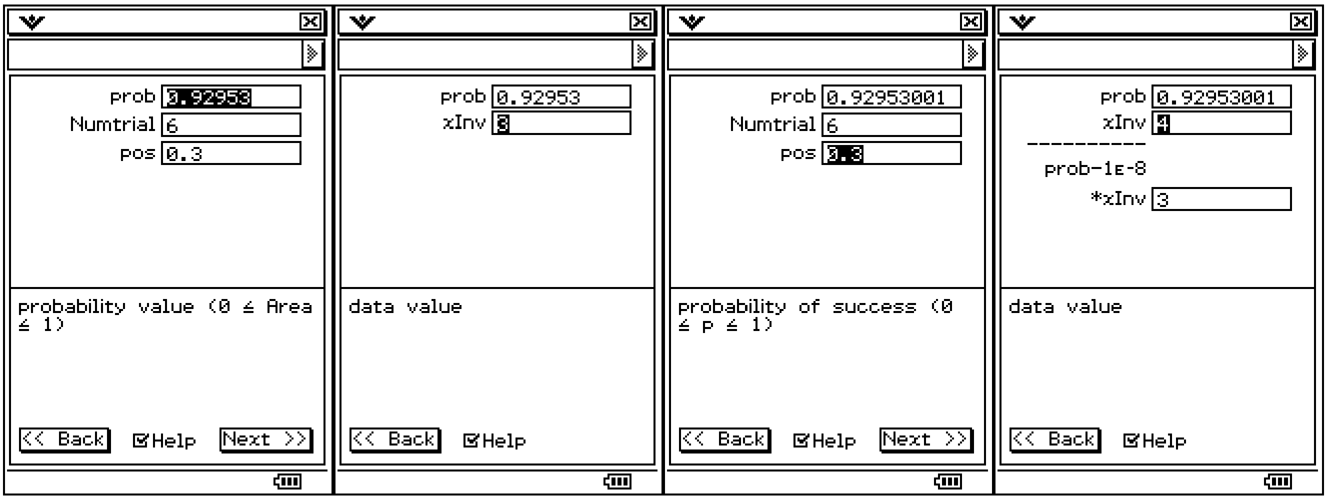


8th step:
Computing of the quantiles of the binomial distribution (ClassPad's solution)

Open the Stat-List-Editor and choose Calc and Distribution. Open Inverse Binomial CD:







The last screenshot shows, that a change of 1E-8 will change the quantile from 4 to 3.