F77 Program 22

```
c       Fortran 77 program to do a simulation of the Weibull distribution
c       of the simplest censoring model -
c                                    x =t if t < Constant
c                                    x =constant if t >= Constant
c       (where Constant=Mean of something, etc)
c       Accelerated life model with 2 treatments. With subroutines.
c       This program may be adapted for other forms of censored models as
c       well as the simplest censoring model.
c       ***************************************************************
c               By Derek Dhammaloka FDX3 - 11th Mar. 1991
c       ***************************************************************
c               Define the following variables
c
c               cdf is the cumulative density function of the probability
c               function and is between 0 and 1. The function urand will
c               generate the random numbers between 0 and 1. It has 1
c               parameter iy, the seed to initialise the generator.
c               t is the remission time in arbitary units
c               kappa is the index to be entered by the user
c               rho is the rate to be entered by the user
c               loop is used in loop counters
c               n is the no. of individuals per treatment to be entered
c               by the user.
c               d is the no. of uncensored individuals in each
c               treatment group.
c               trt is the no. of treatments
c               c is the constant censor time
c               const is the minimum time for censoring to occur. It is
c               to be entered by the user.
c               i is the indicator variable (1 if censored, 0 otherwise)
c               x is equal to t if t is less than C, C otherwise
c               surf is the survivor function
c               hazard is the hazard function
c               pdf is the probability density function
c               ihazard is the integrated hazard
c               a is the acceleration factor to be entered by the user
c               iy is the seed to be entered by the user
c       ***************************************************************
c               Obtain the survivor, hazard and probability density
c               functions. Also the integrated hazard.
c       ***************************************************************
        integer i(2,5000)
        real cdf(2,5000),t(2,5000),c(2,5000),x(2,5000)
        real kappa,const,rho,surf,hazard,pdf,ihazard,a
        integer loop,trt,n,d(2),iy(2)
        trt=2
c       ***************************************************************
c               Input the no. of individuals
c               Also the index (kappa) and the rate (rho)
c       ***************************************************************
        print*,'How many individuals per treatment'
        read*,n
```

```
c
c           Set the no. of failures in each treatment group to the
c           no. of observations per treatment
c
      do 1 loop=1,trt
              d(loop)=n
1     continue
      do 2 loop=1,trt
         print*, 'Seed for the Weibull distribution in treatment  ',loop
         read*,iy(loop)
2     continue
      print*,'Enter the index parameter'
      read*,kappa
      print*,'Enter the rate parameter'
      read*,rho
      print*,'Enter the acceleration factor'
      read*,a
      print*,'Enter the minimum time for censoring to occur'
      read*,const
c
c           Call the subroutine that does the simulation of the
c           Weibull distribution and demonstrates the survivor
c           functions, etc
c
      do 3 loop=1,trt
         if(loop.eq.1) then
          call simulate(kappa,rho,n,cdf,t,c,i,trt,x,const,iy,d,loop)
         else
          call simulate(kappa,rho*a,n,cdf,t,c,i,trt,x,const,iy,d,loop)
         endif
3     continue
      stop
      end

      subroutine simulate(kappa,rho,n,cdf,t,c,i,trt,x,const,iy,d,tn)
c
c           This subroutine simulates the Weibull distribution
c           with rate parameter (rho) and index parameter (kappa).
c           It then decides on whether each individual is to be
c           censored, using the censoring rules.
c           Finally, it prints the survivor function, pdf, hazard
c           function and integrated hazard.
c
c           tn is the treatment number
c
      integer trt,loop,tn,n
      integer iy(trt),d(trt),i(trt,n)
      real kappa,rho,cdf(trt,n),t(trt,n),c(trt,n),x(trt,n)
      real surf,pdf,hazard,ihazard,const
c*****************************************************************
c           Simulate the Weibull distribution
c           using the two parameters to obtain the remission times.
c           However, the censor times are constant.
c*****************************************************************
c           Print the headings
c*****************************************************************
```

```
      print*
      print*,'For treatment ',tn
      print*
      print*,'For the simplest censored model'
      print*
      print*,'Simulation of the Weibull distribution with '
      print*,'Index = ',kappa,' and rate = ',rho
      print*
      print*,'Minimum time for censoring to occur =  ',const
      print*
      write(*,25)
25    format(t3,'time',t17,'I',t25,'sf',t32,'h',t44,'f',t55,'H')
      do 20 loop=1,n
            cdf(tn,loop)=urand(iy(tn))
            surf=1-cdf(tn,loop)
            t(tn,loop)=(-log(surf)/(rho**kappa))**(1/kappa)
            c(tn,loop)=const
c
c           Decide whether the individual is to be censored
c           using the censoring rules
c
            if(t(tn,loop).ge.c(tn,loop)) then
                                    i(tn,loop)=1
                                    d(tn)=d(tn)-1
                                    x(tn,loop)=c(tn,loop)
            endif
            if(t(tn,loop).lt.c(tn,loop)) then
                                    i(tn,loop)=0
                                    x(tn,loop)=t(tn,loop)
            endif
c
c           Recalculate the survivor function
c
            surf=exp(-(rho*x(tn,loop))*kappa)
c
            hazard=(kappa*rho)*((rho*x(loop))**(kappa-1))
            pdf=surf*hazard
            ihazard=((rho*x(loop))**(kappa))
c     ***************************************************************
c           Output the survivor, hazard and density functions as
c           well as the integrated hazard, together with its time
c           and indicator variable
c     ***************************************************************
          write(*,40)x(loop),i(loop),surf,hazard,pdf,ihazard
40        format(f7.3,t12,i7.0,t20,f7.4,t30,f7.4,t40,f7.4,t50,f7.4,t60)
20    continue
      print*
      print*, No. of failures =  ,d(tn)
      return
      end


      real function urand(iy)
      integer iy
```

```
c     ****************************************************************
c           Urand is a uniform random number generator based on
c           theory and suggestions given by KNUTH (1969). The
c           integer iy should be initialised to an arbitary integer
c           prior to the first call to urand. The calling program
c           should not alter the value of iy between subsequent
c           calls to urand. Values of urand will be returned in the
c           interval (0,1).
c     ****************************************************************
c           Reference - Problem solving with Fortran 77
c                       Brian D.Hahn 1987
c     ****************************************************************
      integer ia,ic,itwo,m2,m,mic
      double precision halfm
      real s
      data m2/0/,itwo/2/

c     If first entry, compute machine integer word length
      if(m2.eq.0) then
                  m=1
10    if(m.gt.m2) then
                  m2=m
                  m=itwo*m2
                  goto 10
      endif
      halfm=m2

c     Compute multiplier and increment for linear congruential method
      ia=8*int(halfm*atan(1.d0)/8.d0)+5
      ic=2*int(halfm*(0.5d0-sqrt(3.d0)/6.d0))+1
      mic=(m2-ic)+m2

c     s is the scale factor for converting to floating point
      s=0.5/halfm
      endif

c     Compute next random number
      iy=iy*ia

c     The following statement is for computers which do not allow
c     integer overflow on addition
      if(iy.gt.mic) iy=(iy-m2)-m2
      iy=iy+ic

c     The following statement is for computers where the word length
c     is greater than for multiplication
      if(iy/2.gt.m2) iy=(iy-m2)-m2
c
c     The following statement is for computers where integer overflow
c     affects sign bit
      if(iy.lt.0) iy=(iy+m2)+m2
      urand=float(iy)*s
      return
      end
```

Output from F77 Program 22

```
How many individuals per treatment
175
Seed for the Weibull distribution in treatment                    1
2
Seed for the Weibull distribution in treatment                    2
-1
Enter the index parameter
3
Enter the rate parameter
1.25
Enter the acceleration factor
2
Enter the minimum time for censoring to occur
0.875

For treatment                    1

For the simplest censored model

Simulation of the Weibull distribution with
Index =      .3000000E+01 and rate =      .1250000E+01

Minimum time for censoring to occur =      .8750000E+00
```

time	I	sf	h	f	H
.875	1	.2702	4.4861	1.2121	1.3084
.536		.7401	1.6841	1.2464	.3010
.589		.6714	2.0302	1.3631	.3983
.803		.3639	3.7771	1.3745	1.0109
.639		.6011	2.3906	1.4370	.5090
.485		.8006	1.3763	1.1019	.2224
.444		.8425	1.1570	.9748	.1714
.839		.3156	4.1239	1.3016	1.1532
.761		.4229	3.3932	1.4348	.8607
.538		.7372	1.6990	1.2524	.3049
.624		.6220	2.2825	1.4196	.4749
.703		.5073	2.8961	1.4691	.6787
.333		.9305	.6493	.6042	.0721
.678		.5437	2.6953	1.4655	.6093
.769		.4114	3.4649	1.4255	.8882
.875	1	.2702	4.4861	1.2121	1.3084
.745		.4461	3.2513	1.4503	.8073
.875	1	.2702	4.4861	1.2121	1.3084
.798		.3701	3.7353	1.3822	.9941
.812		.3510	3.8668	1.3571	1.0471
.347		.9219	.7036	.6487	.0813
.655		.5776	2.5139	1.4520	.5489
.807		.3578	3.8191	1.3665	1.0278
.662		.5675	2.5676	1.4570	.5665
.875	1	.2702	4.4861	1.2121	1.3084
.875	1	.2702	4.4861	1.2121	1.3084
.875	1	.2702	4.4861	1.2121	1.3084
.875	1	.2702	4.4861	1.2121	1.3084
.623		.6235	2.2747	1.4182	.4724

```
.875                .2708     4.4810     1.2136     1.3062
.353                .9180      .7284      .6686      .0856
.259                .9666      .3931      .3800      .0339
.684                .5358     2.7383     1.4672      .6240
.618                .6309     2.2365     1.4110      .4606
.875          1     .2702     4.4861     1.2121     1.3084
.822                .3376     3.9619     1.3375     1.0859
.806                .3592     3.8098     1.3683     1.0240
.875          1     .2702     4.4861     1.2121     1.3084
.875          1     .2702     4.4861     1.2121     1.3084
.286                .9554      .4791      .4578      .0457
.797                .3725     3.7188     1.3852      .9875
.617                .6320     2.2312     1.4100      .4589
.604                .6500     2.1388     1.3903      .4307
.875          1     .2702     4.4861     1.2121     1.3084
.730                .4673     3.1254     1.4604      .7609
.519                .7607     1.5800     1.2019      .2735
.396                .8854      .9211      .8155      .1217
.764                .4192     3.4158     1.4320      .8694
.457                .8296     1.2255     1.0167      .1868
.443                .8442     1.1479      .9691      .1694
.875          1     .2702     4.4861     1.2121     1.3084
.875          1     .2702     4.4861     1.2121     1.3084
.594                .6636     2.0696     1.3735      .4100
.875          1     .2702     4.4861     1.2121     1.3084
.702                .5095     2.8837     1.4692      .6744
.394                .8875      .9092      .8069      .1194
.631                .6121     2.3337     1.4284      .4909
.664                .5649     2.5815     1.4582      .5712
.274                .9605      .4405      .4231      .0403
.732                .4655     3.1359     1.4597      .7647
.875          1     .2702     4.4861     1.2121     1.3084
.668                .5580     2.7138     1.5144      .5833
.875          1     .2702     4.4861     1.2121     1.3084
.875          1     .2702     4.4861     1.2121     1.3084
.512                .7692     1.5371     1.1823      .2624
.875          1     .2702     4.4861     1.2121     1.3084
.658                .5733     2.5366     1.4543      .5563
.537                .7394     1.6876     1.2478      .3019
.620                .6273     2.2551     1.4146      .4663
.713                .4930     2.9768     1.4675      .7072
.802                .3649     3.7701     1.3758     1.0080
.875          1     .2702     4.4861     1.2121     1.3084
.875          1     .2702     4.4861     1.2121     1.3084
.749                .4396     3.2902     1.4655      .8218
.875          1     .2702     4.4861     1.2121     1.3084
.310                .9433      .5641      .5322      .0583
.541                .7366     1.7167     1.2594      .3097
.419                .8657     1.0311      .8927      .1442
.875          1     .2702     4.4861     1.2121     1.3084
.814                .3481     3.8872     1.3530     1.0554
.846                .3062     4.1959     1.2847     1.1836
.875          1     .2702     4.4861     1.2121     1.3084
.578                .6855     1.9591     1.3430      .3776
.875          1     .2702     4.4861     1.2121     1.3084
.561                .7080     1.8456     1.3067      .3453
```

.384		.8953	.8642	.7737	.1106
.510		.7722	1.5217	1.1751	.2585
.614		.6365	2.2080	1.4053	.4518
.733		.4635	3.1476	1.4589	.7690
.619		.6286	2.2483	1.4133	.4642
.761		.4229	3.3930	1.4348	.8607
.875	1	.2702	4.4861	1.2121	1.3084
.723		.4784	3.0608	1.4641	.7374
.590		.6699	2.0380	1.3652	.4006
.822		.3377	3.9612	1.3376	1.0857
.627		.6173	2.3063	1.4238	.4823
.690		.5269	2.7871	1.4685	.6407
.712		.4945	2.9685	1.4678	.7043
.366		.9085	.7857	.7139	.0959
.810		.3548	3.8400	1.3624	1.0362
.601		.6548	2.1146	1.3846	.4235
.532		.7448	1.6603	1.2367	.2946
.634		.6075	2.3574	1.4321	.4984
.875	1	.2702	4.4861	1.2121	1.3084
.875	1	.2702	4.4861	1.2121	1.3084
.790		.3811	3.6610	1.3953	.9646
.558		.7127	1.8221	1.2986	.3387
.481		.8049	1.3544	1.0902	.2171
.875	1	.2702	4.4861	1.2121	1.3084
.420		.8648	1.0360	.8960	.1452
.504		.7791	1.4868	1.1583	.2497
.875	1	.2702	4.4861	1.2121	1.3084
.869		.2774	4.4257	1.2279	1.2821
.368		.9070	.7950	.7211	.0976
.521		.7587	1.5902	1.2065	.2761
.804		.3618	3.7916	1.3718	1.0167
.875	1	.2702	4.4861	1.2121	1.3084
.474		.8120	1.3177	1.0699	.2083
.875	1	.2702	4.4861	1.2121	1.3084
.846		.3070	4.1899	1.2862	1.1810
.869		.2782	4.4199	1.2294	1.2796
.300		.9484	.5290	.5017	.0530
.867		.2801	4.4037	1.2335	1.2726
.663		.5666	2.5724	1.4575	.5682
.452		.8349	1.1975	.9998	.1805
.793		.3781	3.6811	1.3919	.9726
.688		.5298	2.7714	1.4682	.6353
.875	1	.2702	4.4861	1.2121	1.3084
.495		.7887	1.4377	1.1339	.2374
.875	1	.2702	4.4861	1.2121	1.3084
.775		.4027	3.5206	1.4176	.9097
.875	1	.2702	4.4861	1.2121	1.3084
.815		.3471	3.8937	1.3517	1.0580
.733		.4627	3.1521	1.4585	.7706
.437		.8496	1.1188	.9506	.1630
.353		.9178	.7296	.6696	.0858
.875	1	.2702	4.4861	1.2121	1.3084
.561		.7086	1.8425	1.3057	.3444
.812		.3513	3.8642	1.3576	1.0460
.812		.3510	3.8666	1.3571	1.0470
.875	1	.2702	4.4861	1.2121	1.3084

.855		.2954	4.2801	1.2644	1.2194
.258		.9669	.3913	.3783	.0337
.875	1	.2702	4.4861	1.2121	1.3084
.850		.3012	4.2343	1.2755	1.1998
.362		.9112	.7698	.7014	.0930
.865		.2824	4.3847	1.2383	1.2644
.432		.8541	1.0946	.9349	.1577
.589		.6707	2.0340	1.3642	.3995
.500		.7837	1.4630	1.1466	.2437
.572		.6670	2.0528	1.3692	.4050
.875	1	.2702	4.4861	1.2121	1.3084
.505		.7780	1.4924	1.1610	.2511
.857		.2920	4.3072	1.2577	1.2310
.536		.7405	1.6822	1.2456	.3004
.426		.8598	1.0634	.9144	.1510
.624		.6223	2.2806	1.4193	.4743
.195		.9857	.2221	.2189	.0144
.734		.4616	3.1586	1.4580	.7730
.820		.3409	3.9381	1.3425	1.0762
.574		.6905	1.9338	1.3353	.3703
.678		.5444	2.6915	1.4653	.6081
.536		.7404	1.6824	1.2457	.3005
.508		.7741	1.5123	1.1706	.2561
.875	1	.2702	4.4861	1.2121	1.3084
.562		.7075	1.8484	1.3077	.3461
.205		.9832	.2474	.2432	.0169
.875	1	.2702	4.4861	1.2121	1.3084
.475		.8110	1.3229	1.0728	.2095
.563		.7056	1.8578	1.3109	.3487
.725		.4746	3.0827	1.4630	.7453
.622		.6249	2.2676	1.4169	.4702
.657		.5751	2.5270	1.4533	.5532
.875	1	.2702	4.4861	1.2121	1.3084
.875	1	.2702	4.4861	1.2121	1.3084

No. of failures = 133

For treatment 2

For the simplest censored model

Simulation of the Weibull distribution with
Index = .3000000E+01 and rate = .2500000E+01

Minimum time for censoring to occur = .8750000E+00

time	I	sf	h	f	H
.478		.1814	10.7131	1.9431	1.7072
.355		.4959	5.9208	2.9360	.7014
.297		.6640	4.1361	2.7462	.4095
.417		.3226	8.1438	2.6268	1.1315
.468		.2014	10.2708	2.0683	1.6026
.491		.1571	11.3062	1.7761	1.8509
.241		.8041	2.7166	2.1845	.2180
.420		.3136	8.2784	2.5961	1.1597
.364		.4712	6.2042	2.9237	.7524

.344	.5289	5.5527	2.9366	.6370
.219	.8477	2.2579	1.9141	.1652
.263	.7528	3.2399	2.4391	.2839
.373	.4452	6.5125	2.8996	.8092
.378	.4304	6.6934	2.8807	.8431
.113	.9779	.5949	.5818	.0223
.304	.6458	4.3205	2.7903	.4372
.378	.4301	6.6963	2.8803	.8436
.301	.6519	4.2587	2.7762	.4279
.357	.4899	5.9890	2.9340	.7136
.408	.3458	7.8070	2.6993	1.0620
.547	.0776	14.0205	1.0882	2.5559
.237	.8118	2.6375	2.1411	.2085
.037	.9992	.0647	.0646	.0008
.265	.7473	3.2957	2.4629	.2913
.396	.3785	7.3577	2.7845	.9747
.575	.0514	15.4905	.7961	2.9683
.318	.6048	4.7425	2.8684	.5028
.280	.7092	3.6796	2.6095	.3436
.474	.1900	10.5179	1.9984	1.6607
.119	.9743	.6586	.6417	.0260
.430	.2887	8.6672	2.5024	1.2423
.402	.3614	7.5886	2.7425	1.0178
.229	.8281	2.4665	2.0425	.1886
.270	.7357	3.4128	2.5108	.3070
.413	.3330	7.9898	2.6608	1.0996
.370	.4535	6.4134	2.9085	.7908
.262	.7548	3.2197	2.4303	.2813
.288	.6884	3.8892	2.6772	.3734
.597	.0359	16.7173	.5997	3.3278
.388	.4001	7.0740	2.8304	.9160
.194	.8928	1.7573	1.5689	.1134
.357	.4920	5.9648	2.9347	.7093
.351	.5075	5.7895	2.9383	.6782
.479	.1800	10.7457	1.9339	1.7150
.320	.5982	4.8117	2.8782	.5139
.535	.0911	13.4283	1.2234	2.3957
.230	.8276	2.4717	2.0456	.1892
.266	.7448	3.3213	2.4736	.2947
.232	.8229	2.5216	2.0750	.1949
.365	.4686	6.2351	2.9218	.7580
.408	.3452	7.8145	2.6978	1.0635
.533	.0939	13.3132	1.2234	2.3650
.355	.8276	2.4717	2.0456	.6983
.326	.5810	4.9916	2.9002	.5430
.388	.4025	7.0438	2.8348	.9102
.294	.6715	4.0562	2.7251	.3977
.296	.6659	4.1167	2.7412	.4067
.395	.3816	7.3153	2.7918	.9633
.440	.2634	9.0891	2.3940	1.3341
.449	.2432	9.4474	2.3979	1.4138
.387	.4039	7.0252	2.8376	.9066
.411	.3375	7.9244	2.6748	1.0861
.344	.5292	5.5486	2.9365	.6363
.186	.9037	1.6289	1.4721	.1012
.575	.0514	15.4908	.7960	2.9684

.408	.3474	7.7842	2.7040	1.0574
.233	.8210	2.5414	2.0865	.1973
.395	.3813	7.3202	2.7909	.9643
.247	.7906	2.8562	2.2580	.2350
.366	.4655	6.2718	2.9193	.7647
.660	.0111	20.4451	.2269	4.5008
.432	.2846	8.7344	2.4856	1.2568
.527	.1013	13.0278	1.3201	2.2894
.411	.3391	7.9024	2.6795	1.0815
.249	.7851	2.9123	2.2864	.2420
.369	.4555	6.3896	2.9105	.7863
.511	.7366	12.2317	1.5239	2.0828
.258	.7646	3.1211	2.3863	.2685
.236	.8144	2.6096	2.1254	.2052
.544	.0809	13.8677	1.1222	2.5143
.366	.4651	6.2761	2.9190	.7655
.438	.2691	8.9911	2.4197	1.3126
.417	.3223	8.1473	2.6260	1.1322
.276	.7206	3.5643	2.5686	.3276
.366	.4651	6.2763	2.9190	.7655
.511	.1241	12.2481	1.5196	2.0870
.417	.3228	8.1407	2.6275	1.1308
.184	.9076	1.5823	1.4362	.0969
.361	.4797	6.1061	2.9291	.7346
.285	.6960	3.8122	2.6534	.3624
.420	.3153	8.2530	2.6019	1.1543
.362	.4769	6.1388	2.9274	.7405
.476	.1858	10.6130	1.9714	1.6833
.340	.5422	5.4066	2.9316	.6121
.278	.7156	3.6155	2.5871	.3347
.434	.2786	8.8316	2.4609	1.2778
.369	.4549	6.3966	2.9099	.7876
.492	.1564	11.3249	1.7709	1.8555
.241	.8039	2.7193	2.1860	.2183
.377	.4332	6.6581	2.8846	.8364
.232	.8225	2.5253	2.0771	.1954
.255	.7708	3.0574	2.3567	.2603
.383	.4722	6.1925	2.9244	.7503
.250	.7827	2.9370	2.2987	.2451
.542	.0835	13.7532	1.1481	2.4832
.513	.1220	12.3125	1.5026	2.1034
.336	.5518	5.3035	2.9263	.5946
.285	.6959	3.8134	2.6537	.3626
.582	.0462	15.8586	.7327	3.0747
.325	.5848	4.9522	2.8959	.5365
.758	.0011	26.9119	.0301	6.7971
.199	.8840	1.8575	1.6421	.1233
.116	.9762	.6257	.6108	.0241
.326	.5820	4.9810	2.8991	.5412
.269	.7378	3.3914	2.5022	.3041
.395	.3813	7.3194	2.7911	.9641
.260	.7601	3.1659	2.4065	.2743
.292	.6765	4.0095	2.7123	.3909
.391	.3932	7.1638	2.8166	.9335
.407	.3485	7.7682	2.7072	1.0541
.417	.3221	8.1503	2.6254	1.1328

.446	.2496	9.3312	2.3294	1.3878
.255	.7724	3.0414	2.3492	.2582
.216	.8547	2.1829	1.8657	.1570
.417	.3229	8.1388	2.6279	1.1304
.481	.1755	10.8498	1.9044	1.7400
.425	.3011	8.4714	2.5504	1.2004
.428	.2940	8.5831	2.5232	1.2243
.293	.6737	4.0372	2.7199	.3949
.390	.3944	7.1479	2.8191	.9304
.451	.2389	9.5264	2.2763	1.4315
.322	.5931	4.8649	2.8853	.5224
.192	.8959	1.7214	1.5421	.1100
.246	.7932	2.8289	2.2439	.2316
.155	.9430	1.1328	1.0682	.0587
.209	.8663	2.0559	1.7810	.1435
.215	.8570	2.1574	1.8490	.1543
.426	.2992	8.5011	2.5432	1.2068
.138	.9602	.8868	.8514	.0407
.293	.6738	4.0359	2.7196	.3948
.178	.9160	1.4808	1.3565	.0877
.323	.5920	4.8762	2.8867	.5242
.454	.2316	9.6639	2.2384	1.4626
.340	.5401	5.4298	2.9326	.6160
.684	.0068	21.9108	.1486	4.9934
.409	.3421	7.8584	2.6887	1.0725
.474	.1896	10.5266	1.9959	1.6628
.197	.8878	1.8147	1.6111	.1190
.286	.6934	3.8387	2.6617	.3662
.396	.3801	7.3361	2.7882	.9674
.612	.0280	17.5384	.4909	3.5760
.584	.0445	15.9898	.7111	3.1129
.273	.7266	3.5045	2.5463	.3194
.282	.7039	3.7329	2.6275	.3511
.168	.9280	1.3303	1.2345	.0747
.362	.4775	6.1314	2.9278	.7392
.505	.1339	11.9473	1.6000	2.0105
.272	.7308	3.4615	2.5298	.3135
.347	.5204	5.6462	2.9382	.6532
.179	.9138	1.5075	1.3776	.0901
.398	.3741	7.4163	2.7742	.9833
.284	.6994	3.7785	2.6425	.3576
.401	.3647	7.5428	2.7512	1.0086
.354	.5007	5.8665	2.9372	.6918
.324	.5870	4.9285	2.8931	.5327
.310	.6276	4.5073	2.8287	.4659
.273	.7283	3.4875	2.5398	.3171
.297	.6637	4.1390	2.7469	.4100
.364	.4711	6.2061	2.9236	.7527
.270	.7349	3.4202	2.5137	.3080
.373	.4441	6.5262	2.8983	.8117
.234	.8183	2.5693	2.1025	.2005
.234	.8182	2.5703	2.1031	.2006
.271	.7328	3.4416	2.5221	.3109
.507	.1312	12.0278	1.5783	2.0309

No. of failures = 175