Microview is an alternate monthly column for computer game reviews and game-aid programs, edited by Russell Clarke.

GROW YOUR OWN PLANETS

ACRETE, by Steve Gilham is a gigantic program of Cosmic evolution that produces results compatible with current day science. It is written in Microsoft Basic *(actually on an IBM PC), but is readily convertible to other machines that have more than thirty column displays using the notes that accompany the program.*

The program accompanying this article is a simplified version of a program called ACRETE designed by Dr Stephen Dole of the Rand Corporation, the results of which were published in the astronomical journal *Icarus* in 1970. It generates solar systems. It takes the planets that it has generated, and works out the most obvious properties, such as the year, the temperature, etc. The information used to derive these quantities is either standard theory, or can be found in Dr Dole's book *Habitable Planets for Man* (*Elsevier Press*, 1970).

The program grows planets starting from a cloud of gas and dust circling a central star. One at a time small bodies – asteroid size – are introduced into the dustcloud in elliptic orbits. The body sweeps up all the dust crossing its orbit, and any nearby dust which its gravity can bring to it. If it grows large enough, it can also capture the gas (hydrogen and helium) from its vicinity. The program then assumes that once gas starts to be captured, an amount of gas usually equal to 25 times the mass of dust is captured.

When the planet has stopped growing, another one is started. If two planets should have orbits that cross, they are assumed to collide, and the new planet allowed to grow by its gravitational attraction. Like the original ACRETE program, this program does not generate moons, or determine atmospheres.

I wrote the program on an *IBMPC* with 128k store, but the actual code only takes up 6k when stored on disk, so it should be usable on most machines with 16k or more. The symbol ^ denotes taking powers, and EXP and LOG are the mathematical functions e^x where e=2.718... and natural logarithms (ln).

What the Program does

Lines 10-20 clear the screen and put it into high-resolution graphics mode. Line 30 initializes the random number generator. Lines 40-100 input the basic details of the system. Lines 110-190 actually put the dust into the cloud, each element of the array holding the dust in a band 0.1 AU wide (1AU is the radius of the Earth's orbit). A range of orbits up to 50AU out is allowed (compare Pluto at just over 30AU). The program works internally using a mass equal to 1.OE-4 of the Sun's as a unit, but all output is in terms of the Earth's mass (0.03 of the internal unit). LIne 200 works out how luminous the central star is, in terms of the Sun's brightness. This assumes that the star is a normal or 'Main Sequence' star, not a white dwarf or a giant.

The main loop that grows a planet is in lines 250-400; lines 280-330 being an optional way of checking how the system is growing while the program is working. The array SWP holds details of planet N as follows:

SWP (N,O) – mass of dust in the planet, in internal units.
SWP (N,1) – inner edge of its influence, in AU.
SWP (N,2) – semi-major axis of its orbit (the orbit is elliptical, and the semi-major axis is half the length of the longest diameter) also in AU.
SWP (N,3) – eccentricity of the orbit (how elliptical it is – 0 means a circle, 1 a straight line); most planetary orbits have eccentricities below 0.1.
SWP (N,4) – outer edge of its influence in AU.
SWP (N,5) – total mass, gas and dust, in the planet, in internal units.

The string array V$(N) holds 'r' to denote a rocky or icy planet like the Earth, or Mars, or Pluto, and 'g' for a gas-giant. Any planet that has begun to gather hydrogen gas from the cloud is assumed to be a gas-giant. Gas-giants above 6700 times the Earth in mass are large enough to become stars, and are denoted by '*'.

When all the planets are grown, they are sorted into order of distance from the central star (line 410) and then interpreted (lines 430-490), and finally a schematic picture is drawn (line 500) before restoring the screen to text mode.

The subroutine from line 1000 is just there to clear the screen between sets of output, so it's easier to dump the screen to the printer. The first significant subroutine is at line 2000. This checks whether the current planet has come close enough to any others to have collided. If it does, it determines the resulting orbit, and then checks whether this body will collide with any others, finishing when there have been no collisions. If there is a collision, it checks if the new planet can grow further by calling the accretion subroutine.

This subroutine in lines 2300-2470 is the key part of the program. It checks how widely the planet sweeps in its orbit (line 2320), and adds a correction for the planet's gravity (lines 2350-2360). MC in line 2330 is the mass above which hydrogen can be captured. As dust is captured (lines 2400-2420) the planet grows, and the dust cloud is depleted.

The subroutine from line 2700 uses the graphics functions to give a rough idea of the system. The scale is logarithmic for clarity, and shows the planets as circles with radius proportional to the cube root of their mass. The different types are shown in different colours.

The Inputs to the ACRETE Program

The inputs required by the program start with the seed for the random number generator (line 30) – this may vary from computer to computer – followed by a description of the system under consideration. The values required are:

1. Maximum eccentricity of the orbits of the bodies we insert to grow planets from. The smaller this value, the narrower the bands swept by the growing planets, and hence there will be more, smaller planets. Larger values mean fewer, bigger planets.
2. Mass of the star in terms of the Sun's – this should be between 0.6 and 1.5 for there to be habitable planets. The program should cope with stars up to 5 times the mass of the Sun. Stars heavier than 1.5 times the Sun burn out their fuel in much less time than it would take life to evolve, and smaller stars than 0.6 times the Sun are too faint for planets to be warm enough without being slowed until one face always points towards the star, except in special circumstances.
3. Amount of rocky material available for planet formation (usually 1).
4. Amount of gas available, usually 1. The program takes into account the factor of 25 mentioned earlier.

The program assumes that if no answer is given a value of 0 is understood, and if the value 0 is entered, it assumes a star of the same mass as the Sun, with the normal amounts of gas and dust, and a particle eccentricity of 0.4.

The Output

The output lists the planets in the order of their distance from the central star.

The first screen shows (for reference) the input values, the orbital radius (in AU), the mass of the planet in terms of the Earth, the eccentricity of the planet's orbit, and the type – r for a rocky or icy planet, g for a gas-giant, * for a star.

Screen 2 gives the time taken for one of the planet's orbits in terms of the Earth's year, the mean temperature in Celsius (the Earth has mean temperature 15°C), and the inclination of the planet's axis to the plane of its orbit (23.5° for the Earth). For temperatures above 40°C a planet must have a large inclination angle (above 45°) to be habitable, and will in any case have extreme seasons. I've not found a properly scientific way of generating the actual temperature range on the planet, but if you wish, there are always the tables in *Traveller* book 6, or in *Universe*. In any case, moons will have to come from one of those sources.

Screen 3 gives the radius of the planet and the resultant gravity (Earth has a gravity of 9.81 m/s 2), and a notation for the Size of the planet. The integer part of the number is the value for use in the game *Universe*, (my particular favourite) the fractional part is 1/1000 times the *Traveller* value (in decimal).

The final screen (computed in lines 4350-4440) gives the value of the tidal effect of the star upon the planet, which will affect the length of the planet's day, the value of the day if no tidal influence occurred, and a guide to whether the planet might be habitable. 'No' means what it says, '??' is explained below, and '?' depends on the correct amount of water and oxygen, and on the precise values of the mean temperature and planetary inclination.

With this model the Earth's length of day would be given as > 14.5 hours. The actual length of the day is greater because the Moon and Sun have slowed the spin down. The unit of this tidal slowing, I've taken to be equal to the Moon's effect on the Earth. The Sun only has 0.2 as much effect. If the total tide acting on a planet is much above 1.5 the day will be very long, and the nights will be much colder than the days, leaving the planet not very habitable, if the sun's effect is dominant. If a planet's moons dominate its tides, and it is slowed by them, then it may keep one face facing its largest moon, just as the Moon faces the Earth. In that case, if the moon is close enough that its orbit is less than about 4 days, then the planet can be habitable even with strong sun tides. The upper limit comes when the actual ocean tides become too destructive. Planets marked '??' under the suitability column would need a large close moon like this to be habitable. A similar type of planet could also exist as the moon of a gas-giant in the otherwise habitable temperature zone (-14 to 65°C).

As an example of the use of the program I've included some sample output. This run takes the usual values of the mass of the star, the amount of dust, and so on. We have a system of 8 planets, two small ones close to the star, one probably like Venus, a large planet which gets a '?' result, and 4 gas-giants. Planet 4 has a high tilt, and is cooler than the Earth, so will be somewhat Arctic in nature, as if the Tropics had been removed, and the rest of the Earth's climates moved down a bit. With just over twice the mass of the earth, the gravity is fairly high – about 1.4g – and for *Traveller*, the planet is size 9. The day is anything above 13 hours long, depending on moons.□

Sample Print-Out

Max e: .4 Stellar mass: 1
Dust: 1 Gas: 1

Planet	*Orbit*	*Mass*	*e*	*Type*
1	.1	.048	.292	r
2	.2	.105	.048	r
3	.5	1.14	.33	r
4	1.055	2.033	.188	r
5	2.41	141.647	.361	g
6	6.488001	276.755	.248	g
7	17.27	173.939	.234	g
8	33.295	20.217	.024	g

Planet	*Year*	*Temp c*	*Incl*
1	.031	637	102.6
2	.089	370	38.1
3	.353	134	8.899999
4	1.083	7	65.2
5	3.741	–88	81.2
6	16.525	–160	30.2
7	71.76901	–204	9.8
8	192.118	–224	31.8

Planet	*Radius (km)*	*g(m/s^2)*	*Size*
1	2658	2.728	2.003
2	3363	3.729	2.004
3	6633	10.409	5.008
4	7719	13.707	6.009
5	47953	24.751	9.059
6	62043	28.889	9.076999
7	51894	25.952	9.064
8	22679	15.792	8.028

Planet	*Sun tide*	*Day > (hrs)*	*Suitable*
1	174853.1	29.198	no
2	3742.981	24.972	no
3	29.781	14.947	no
4	.357	13.026	?
5	1.147	9.693001	no
6	.006	8.972	no
7	0	9.466001	no
8	0	12.135	no

Star Types and Masses

	Mass (Sun = 1)		
Spectral Type	*Average*	*Range*	*Percentage Die-Roll*
M	0.22	0.2-0.5	01-72
K	0.6	0.5-0.75	73-87
G	0.9	0.75-1.02	88-94
F	1.25	1.02-1.55	95-97
A	2	1.55-3	98
B	6	3 -10	99(01-90)
O	32	10 -60	99(91-00)

A roll of 00 means a non-Main Sequence Star, either a red giant, which will have swollen to engulf its innermost planets, or a white dwarf, which is the next stage of a star's life after being a red giant.

Dealing with either of these cases is beyond the scope of the ACRETE program.

Stars above 60 solar masses are unstable, and are unlikely to have planets. Other evidence means that it is unlikely that stars above 1.25 solar masses have planets. Stars may exist down to 0.02 solar masses, but they are generally too faint to observe, so we don't know how frequent they are below 0.2 solar masses (M5 type).

```
10    CLS
20    SCREEN 2
30    RANDOMIZE
40    DIM STO (500), SWP (25,5), V$(25), HLD(25,5),U$(25)
50    INPUT "max e ",E: INPUT "stellar mass ",MS:INPUT
      "dust ",D:INPUT "gas ",G
60    IF E=0 THEN E =.4
70    IF MS = 0 THEN MS = 1
80    IF D=0 THEN D=1
90    G=G*25: IF G=0 THEN G=25
100   G=G/D
110   FOR I=1 TO 500
120   R=(I/10)/(MS^.33)
130   U=R^.33)
140   STO(I)=D*1.5*R*R*EXP(-5!*U)
150   NEXT!
160   SM=STO(500)
170   IN=3*SQR(LS):IN=INT(IN): IF IN <.5 THEN IN = 1
180   IF STO(IN) < SM THEN SM=STO(IN)
190   SM=SM/2
200   LS=MS^4:IF MS < .4 THEN LS = .23*(MS^2.3)
210   N=0
220   REM
230   REM start iteration
240   REM
250   GOSUB 2500
260   SWP (N,3)=E*RND:SWP(N,0)=0:V$(N)="r"
270   SWP (N,5)=0
280   IF SWP (N,3) > 1 THEN SWP(N,3)=1
310   FOR KK = 1 TO N-1
320   PRINT KK,SWP(KK,2),SWP(KK,5)/.03
330   NEXT KK
360   K=N
370   GOSUB 2300
380   GOSUB 2000
390   IF N = 26 THEN GOTO 410
400   GOTO 250
410   GOSUB 2900
420   GOSUB 1000
430   PRINT "max e ",E,"stellar mass ",MS:PRINT "dust ",
      D,"gas ",G*D/25
440   PRINT " " :PRINT "#","orbit","mass","e","type"
450   FOR I=1 TO N
460   PRINT I,SWP(I,2),SWP(I,5)/.03,SWP(I,3),V$(I)
470   NEXT I
480   GOSUB 1000
490   GOSUB 4000
500   INPUT "picture", PIC
510   GOSUB 2690
520   INPUT "winddown", PIC
530   SCREEN O: WIDTH 80: STOP
1000  REM------------------
1010  REM next screen
1020  INPUT "next screen",PIC:CLS:RETURN
2000  REM------------------
2010  REM collisions and aftermath
2020  IF N > 1.5 THEN RETURN
2030  FOR K = 1 TO N-1
2040  IF SWP(N,4) < SWP(K,1) THEN GOTO 2210
2050  IF SWP(K,4) < SWP(N,1) THEN GOTO 2210
2060  PRINT "COLLISION" K "AND" N
2070  H=SWP (N,5) *SQR(SWP(N,2)*(1-SWP(N,3)*
      SWP(N,3)))
2080  H=H+SWP(K,5)*SQR(SWP(K,2)*(1-SWP(K,3)*
      SWP(K,3)))
2090  SWP(K,5)=SWP(N,5)+SWP(K,5):H=H/SWP(K,5)
2100  SWP(K,0)=SWP(N,0)+SWP(K,0)
2110  KF=SWP(N,3):IF KF > SWP(K,3) THEN KF=SWP(K,3):
      SWP(K,3)=RND*KF
2120  SWP(K,2)=H*H/(1-SWP(K,3)*SWP(K,3))
2130  GOSUB 2300
2140  N=N-1
2150  IF K=N THEN GOTO 2020
2160  FOR L=0 TO 5:SWP(N+1,L)=SWP(K,L):NEXT L:V$
      (N+1)=V$(K)
2170  FOR L=0 TO 5:SWP(K,L)=SWP(N,L):NEXT L:V$(K)=
      V$(N)
2180  FOR L=0 TO 5:SWP(N,L)=SWP(N+1,L):NEXT
      L:V$(N)=V$(N+1)
2190  FOR L=0 TO 5:SWP(N+1,L)=0:NEXT L:V$
      (N+1)=""
2200  GOTO 2020
2210  NEXT K
2220  RETURN
2300  REM----------------
2310  REM accretion
2320  RA=SWP(K,2)*(1+SWP(K,3)):RP=SWP(K,2)
      *(1-SWP(K,3))
2330  MC=.12*(RP^-.75)*(LS^.375)
2340  SWP(K,5)=SWP(K,0):IF SWP(K,0) > MC
      THEN SWP (K,5) = MC +G*(SWP(K,0)-MC)
2350  KF=.1*((SWP(K,5/MS)^.25)
2360  SWP(K,1)=RP-KF:SWP(K,4)=RA+KF
2370  IM=INT(10*SWP(K,1)+.5):IX=INT(10*SWP
      (K,4)+.5)
2380  IF IM < IN THEN IM=IN
2390  IF IX > 500 THEN IX = 500
2400  DM=0
2410  FOR I=IM TO IX:DM=DM+STO(I):STO(I)=0:
      NEXT I
2420  SWP(K,0)=SWP(K,0)+DM
2430  IF DM > SM/2 THEN GOTO 2340
2440  IF SWP(K,0) > MC THEN V$(K)="g"
2450  SWP(K,5)=SWP(K,0):IF SWP(K,0) > MC
      THEN SWP(K,5) = MC +G*(SWP(K,0)-MC)
2460  IF SWP(K,5) >200 THEN V$(K)="*"
2470  RETURN
2480  REM----------------
2490  REM position of next nucleus
2500  SUM=0
2510  PRINT " "
2520  FOR I=IN TO 500
2530  SUM=SUM+STO(I)
2540  REM X=70+90*LOG(I-.5):Y=70+90*LOG(I+.5)
2550  REM IF STO(I)>0 THEN LINE (X,100)-(Y,100)
2560  NEXT I
2570  PRINT SUM/.03 "LEFT"
2580  IF SUM < SMTHEN GOTO 410
2590  SUM=SUM*RND
2600  FOR I=IN TO 500
2610  SUM=SUM-STO(I)
2620  R=I/10
2630  IF SUM < 0 THEN GOTO 2650
2640  NEXT I
2650  N=N+1
2660  SWP(N,2)=R
2670  RETURN
2680  REM----------------
2690  REM picture
2700  CLS
2710  SCREEN 1
2720  SCL=300/LOG(50/.3):KK=10-SCL*LOG(.3)
2730  LINE (10,100)-(310,100)
2740  FOR I = 1 TO N
2750  IF SWP(I,2) < .3 THEN GOTO 2860
2760  X=KK+SCL*LOG(SWP(I,2))
2770  RA=SWP(I,2)*(1+SWP(I,3)):RP=SWP(I,2)*
      (1-SWP(I,3))
2780  Y=KK+SCL*LOG(RP):Z=KK+SCL*LOG(RA)
2790  IF SWP(I,5) < SM THEN GOTO 2860
2800  R=10*(SWP(I,5)^(1/3))
2810  LINE (Y,140+I)-(Z,140+I):COL=1
2820  MC=.12*(RP^-.75)*(LS^.375)
2830  IF SWP(I,5) > MC THEN COL = 2
2840  IF SWP(I,5) > 200 THEN COL=3
2850  CIRCLE (X,100),R,COL
2860  NEXT I
2870  RETURN
2880  REM----------------
2890  REM sorting
2900  FOR I=1 TO N: M=0 R=55:
2910  FOR J=1 TO N: IF SWP(J,2)>R THEN GOTO
      2950
2920  IF SWP(J,5) < SM/2 THEN GOTO 2940
2930  R=SWP(J,2):M=J:GOTO 2950
2940  N=N-1
2950  NEXT J
2960  FOR L = 0 TO 5
2970  HLD(I,L)=SWP(M,L):NEXT L
2980  U$(I)=V$(M)
2990  SWP(M,2)=60:NEXT I
3000  FOR I = 1 TO N
3010  V$(I)=U$(I)
3020  FOR L = 0 TO 4
3030  SWP(I,L)=.001*INT(1000*HLD(I,L))
3040  NEXT L
3050  SWP(I,5)k=.00003*INT(1000/.03*HLD(I,5))
3060  U$(I)="?":IF V$(I)<>"r" THEN U$(I)="NO"
3070  IF SWP(I,5) < 9/1000 THEN U$(I)="no"
3080  NEXT I
3090  RETURN
4000  REM----------------
4010  REM secondary quantities
4020  PRINT "#","YEAR","temp c","incl."
4030  FOR I=1 TO N
4040  YR=SQR(SWP(I,2)/MS)*SWP(I,2):YR=.001*
      INT(1000*YR)
4050  T=288*(LS^.25)/SQR(SWP (I,2)) -273: T=INT(T)
4060  IN=180*(1-(RND^(2/9))):IN=.1*INT(10*IN)
4070  IF T < -14 THEN U$(I)="no"
4080  IF T > 65 THEN U$(I)="no"
4090  PRINT I,YR,T,IN
4100  NEXT I
4110  GOSUB 1010
4120  REM----------------
4130  PRINT "#","radius(km)","g(m/s^2)","size"
4140  FOR I= 1 TO N
4150  IF V$(I)<>"r" THEN GOTO 4250:REM
      terrestrials
4160  R=(SWP(I,5)/.03)^(1/3):MM=R*(2^(1/3))
4170  RN=MM*2(-R/3)
4180  IF ABS(R-RN) < .01 THEN GOTO 4200
4190  R=RN:GOTO 4170
4200  SWP(I,1)=RN:R=INT(6400*RN):S=
      INT(5.672*RN)
4210  IF S>7 THEN S=7
4220  G=327*SWP(I,5)/(RN*RN):G=.001*INT
      (1000*G)
4230  S=S+INT(8*RN)/1000
4240  PRINT I,R,G,S:GOTO 4320
4250  IF V$(I)<>"g" THEN GOTO 4300:REM gas
      giants
4260  S=8:IF SWP(I,5) > 1 THEN S=9
4270  IF SWP(I,5) > 10 THEN S=10
4280  RN=4.2952*(SWP(I,5)^.3846):IF RN < 10.5
      THEN RN=10.5
4290  SWP(I,1)=RN:R=INT(6400*RN):GOTO 4220
4330  REM stars
4310  RN=.6946*(SWP(I,5)^5936(:S=11: GOTO 4290
4320  NEXT I
4330  REM----------------
4340  GOSUB 1010
4350  PRINT"#","sun tide","day > (hrs)","suitable"
4360  FOR I=1 TO N
4370  H=.020214*MS*(SWP(I,1)^4)/(SWP(I,5)*
      (SWP(I,2)^3)):H=H*H
4380  H=.001*INT(1000*H)
4390  IF H > 2 AND U$(I) = "?" THEN U$(I)="??"
4400  IF H > 400 THEN U$(I)="no"
4410  DY=2.667*SWP(I,1)/SQR(SWP(I,5))
4420  DY=.001*INT(1000*DY)
4430  PRINT I,H,DY,U$(I)
4440  NEXT I
4450  RETURN
```