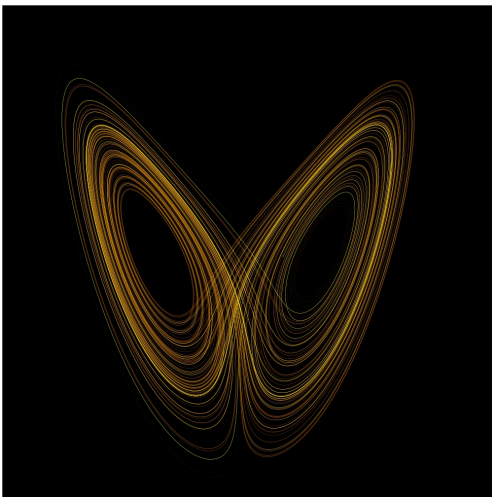


Chaos

Chaos theory is a field of study in mathematics, with applications in several disciplines including meteorology, physics, engineering, economics and biology. It studies the behavior of dynamical systems that are highly sensitive to initial conditions, an effect which is popularly referred to as the *butterfly effect*. Small differences in initial conditions (such as those due to rounding errors in numerical computation) yield widely diverging outcomes for such dynamical systems, rendering long-term prediction impossible in general. This happens even though these systems are deterministic, meaning that their future behavior is fully determined by their initial conditions, with no random elements involved. In other words, the deterministic nature of these systems does not make them predictable.



A plot of the Lorenz attractor for values $r = 28$, $\sigma = 10$, and $b = \frac{8}{3}$.

This behavior is known as *deterministic chaos*, or simply chaos. It was summarised by Edward Lorenz as follows:

The fluttering of a butterfly's wing in Rio de Janeiro, amplified by atmospheric currents, could cause a tornado in Texas two weeks later.

— Edward Lorenz

The following simple model for population growth might be applied to study fish in a pond, bacteria in a test tube, or any of a host of similar situations. We suppose that the population ranges from 0 (extinct) to 1 (maximum population that can be sustained). If the population at time t_i is d , then we suppose the population at time t_{i+1} to be $rd(1-d)$, where the argument r — known as the *fecundity parameter* — controls the growth rate.

Input

Three lines containing *i*) the initial density $d \in \mathbb{R}$ of a population, *ii*) the fecundity parameter $r \in \mathbb{R}$, and *iii*) the number of time steps $s \in \mathbb{N}_0$ at which we wish to simulate the population density (including time point t_0).

Output

s lines containing the population density at time points t_0, t_1, \dots, t_{s-1} .

Example

If r has a value of around 2 , the population density will approach $1 - \frac{1}{r}$ as time goes by.

Input:

0.1
1.9
6

Output:

0.1
0.171
0.26934210000000003
0.373914173018421
0.44479449204530547
0.4692094685937818

Example

If $r = 3$ the population density will converge to a state where the population density alternates between two values.

Input:

0.1
3
12

Output:

0.1
0.27000000000000001
0.59130000000000002
0.7249929299999999
0.5981345443500454
0.7211088336156269
0.603332651091411
0.7179670896552621
0.6074710434816448
0.7153499244388992
0.6108732301324811
0.7131213805199696

Example

If $r \geq 4$ the model will act *chaotic*. This means that the results can change drastically when the initial conditions are slightly perturbed. See the difference in behaviour between example 3 and example 4.

Input:

0.1
4
60

Output:

0.1
0.36000000000000004
0.9216
... (54 lines) ...
0.977464119602946
0.08811205796713474
0.32139329283172413

Example

A difference of 0.00000000001 in the initial population density causes a huge difference in the population density difference on time point t_{59} . This is *chaotic behaviour*, showing that it is hard to predict population densities, because tiny perturbations cause large differences in the long run.

Input:

0.10000000001
4
60

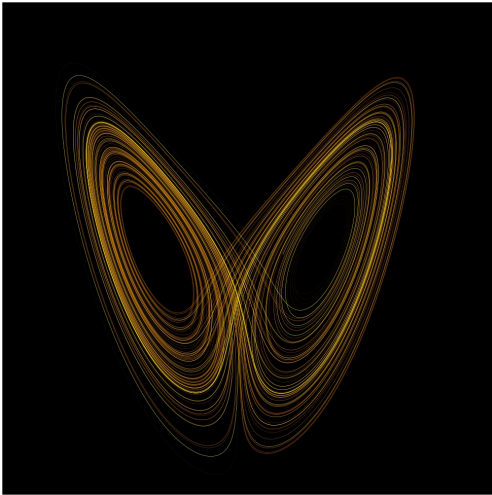
Output:

0.10000000001
0.36000000003200006
0.9216000000358401
... (54 lines) ...
0.830632498181969
0.5627286045838009
0.9842604886678766

Remark: You may get different results for the final two examples. Different programming languages or computing environments can have different ways to round floating point numbers. These tiny differences cause huge differences in processing the floating point numbers due to the *chaotic nature* of the model. As an additional exercise, you can study the difference between using the formula $rd(1-d)$ and the identical formula (mathematically speaking) $rd - rd^2$. Implementations relying on either of these formulas will result in different rounding errors. The meteorologist [Edward Lorenz](#) also noticed these effects caused by rounding errors when studying his [model of atmospheric convection](#). His studies of these effects made him one of the pioneers of *chaos theory*.

De chaostheorie is een onderzoeksdomein binnen de wiskunde die toepassingen heeft in weersvoorspelling, fysica, ingenieurswetenschappen, economie en biologie. Ze onderzoekt het

gedrag van dynamische systemen die extreem gevoelig zijn voor kleine schommelingen in de begintoestand. Een effect dat beter bekend is onder zijn populaire Engelse benaming: *butterfly effect*. Kleine verschillen in de begintoestand van een dynamisch systeem (bijvoorbeeld veroorzaakt door afrondingsfouten bij numerieke berekeningen) kunnen immers een enorme impact hebben op het gesimuleerde gedrag van dergelijke systemen, waardoor voorspellingen op lange termijn schier onmogelijk worden. Dit komt zelfs voor wanneer deze systemen deterministisch zijn, wat erop neerkomt dat het gedrag van het systeem in de toekomst volledig afhangt van de begintoestand, zonder dat er willekeurige invloeden optreden. Met andere woorden, de deterministische aard van dergelijke systemen zorgt er niet voor dat ze voorspelbaar worden.



De Lorenz-attractor van Edward Lorenz in een driedimensionale faseruimte. De bijbehorende differentiaalvergelijkingen beschrijven een eenvoudig meteorologisch model. Het systeem beweegt langs de getoonde banen links met de klok mee, na oversteek rechts tegen de klok in, na oversteek links, enzovoorts. Bij elke omloop verbreedt de band van de banen zich. De figuur is berekend met de standaard parameters $r=28$, $s=10$, en $b = \frac{8}{3}$ (of 2.666667).

Het begrip chaos krijgt hierdoor een technische betekenis — te onderscheiden van het losse alledaagse woordgebruik — en betekent dan dat de schijnbare wanorde toch exact bepaald is en geordend tot stand komt volgens een algoritme of rekenregel. Dit gedrag staat bekend als *deterministische chaos*, of kortweg chaos. Dit wordt door Edward Lorenz kort samengevat op de volgende manier:

The fluttering of a butterfly's wing in Rio de Janeiro, amplified by atmospheric currents, could cause a tornado in Texas two weeks later.

— Edward Lorenz

Het volgende eenvoudige model voor de groei van een populatie kan bijvoorbeeld toegepast worden op het visbestand in een meer, bacteriën in een proefbuis, of tal van andere gelijkaardige omstandigheden. Hierbij wordt verondersteld dat een populatie kan schommelen tussen 0 (volledig uitgestorven) en 1 (maximale populatie die door de omgeving getolereerd wordt). Als de grootte of de dichtheid van de populatie op tijdstip t_i gelijk is aan d , dan veronderstellen we dat de populatie op tijdstip t_{i+1} gelijk is aan $rd(1-d)$, waarbij het argument r — de *vruchtbaarheidsparameter* genoemd — de mate van groei bepaalt.

Invoer

Drie regels die achtereenvolgens de volgende informatie bevatten *i*) de initiële populatiedichtheid $d \in \mathbb{R}$, *ii*) de waarde van de vruchtbaarheidsparameter $r \in \mathbb{R}$, en *iii*) het aantal tijdstappen $s \in \mathbb{N}_0$ waarover we de populatiedichtheid willen simuleren (inclusief het tijdstip t_0).

Uitvoer

s regels die de populatiedichtheid bevatten op de tijdstippen t_0, t_1, \dots, t_{s-1} .

Voorbeeld

Als r een waarde heeft die rond 2 ligt, dan benadert de populatiedichtheid na verloop van tijd de waarde $1 - \frac{1}{r}$.

Invoer:

0.1
1.9
6

Uitvoer:

0.1
0.171
0.26934210000000003
0.373914173018421
0.44479449204530547
0.4692094685937818

Voorbeeld

Als $r = 3$ convergeert de populatiedichtheid naar een toestand waarbij alternerend tussen twee waarden gesprongen wordt.

Invoer:

0.1
3
12

Uitvoer:

0.1
0.27000000000000001
0.59130000000000002
0.72499292999999999
0.5981345443500454
0.7211088336156269
0.603332651091411
0.7179670896552621
0.6074710434816448
0.7153499244388992

0.6108732301324811
0.7131213805199696

Voorbeeld

Als $r \geq 4$ dan wordt het gedrag van het model *chaotisch*. Dat betekent dat de resultaten extreem kunnen verschillen bij zeer kleine schommelingen van de begintoestand. Vergelijk bijvoorbeeld het verschil in het gedrag tussen voorbeeld 3 en voorbeeld 4.

Invoer:

0.1
4
60

Uitvoer:

0.1
0.36000000000000004
0.9216
... (54 regels) ...
0.977464119602946
0.08811205796713474
0.32139329283172413

Voorbeeld

Een verschil van 0.0000000001 in de initiële populatiedichtheid resulteert in een groot verschil in de populatiedichtheid op tijdstip t_{59} . Dit is *chaotisch gedrag*, en het voorbeeld toont aan dat de populatiedichtheid extreem moeilijk te voorspellen is omdat kleine verschillen bij aanvang op termijn kunnen resulteren in grote verschillen.

Invoer:

0.1000000001
4
60

Uitvoer:

0.1000000001
0.36000000003200006
0.9216000000358401
... (54 regels) ...
0.830632498181969
0.5627286045838009
0.9842604886678766

Opmerking: Het zou goed kunnen dat de resultaten die je krijgt voor de laatste twee voorbeelden sterk afwijken van de resultaten die hier getoond worden. Verschillende programmeertalen en rekenomgevingen gebruiken immers verschillende technieken om *floating*

point getallen af te ronden. Deze kleine verschillen kunnen grote verschillen veroorzaken bij het rekenen met floating *point getallen*, te wijten aan de *chaotische eigenschappen* van het model. Als bijkomende oefening (niet verplicht) moet je maar eens proberen om bij je implementatie de formule $rd(1-d)$ te vervangen door de identieke formule (althans wiskundig gezien) $rd - rd^2$. De resultaten van deze implementaties kunnen sterk verschillen naargelang de ene dan wel de andere formule gebruikt wordt, aangezien de formules resulteren in andere afrondingsfouten. De meteoroloog [Edward Lorenz](#) merkte deze effecten die veroorzaakt worden door afrondingsfouten op bij het bestuderen van een [model voor atmosferische convectie](#). Door zijn onderzoek naar deze effecten werd hij de grondlegger van de *chaostheorie*.