Outline

1. Simulation
   - Monte Carlo methods
   - volumes and expected values
2. Repeat Until
   - binary expansion
   - break statement

MCS 260 Lecture 12
Introduction to Computer Science
Jan Verschelde, 26 June 2023

Running Simulations
repeat until: break

1. Simulation
   - Monte Carlo methods
   - volumes and expected values

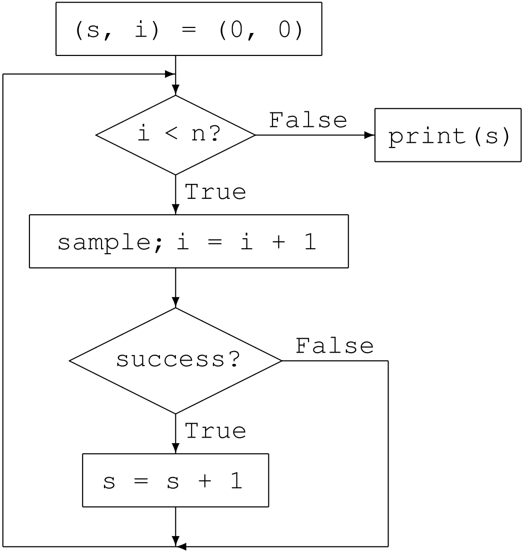
2. Repeat Until
   - binary expansion
   - break statement

Simulation

Monte Carlo methods

- In a mathematical model with uncertainties,
  events occur with assigned probabilities.
- Simulation consists in the repeated drawing of samples according to a probability distribution.
  We count the number of successful samples.
- The Law of Large Numbers states that the arithmetic average of the observed successes converges to the expected value or mean of the experiment, as the number of experiments increases.
- Monte Carlo methods are listed among
  the Top Ten Algorithms of the 20th century.

flowchart for simulations



Running Simulations
repeat until: break

1. Simulation
   - Monte Carlo methods
   - volumes and expected values

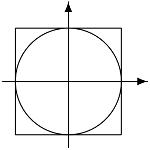
2. Repeat Until
   - binary expansion
   - break statement

Estimating Areas and Volumes

applies to high dimensional integrals

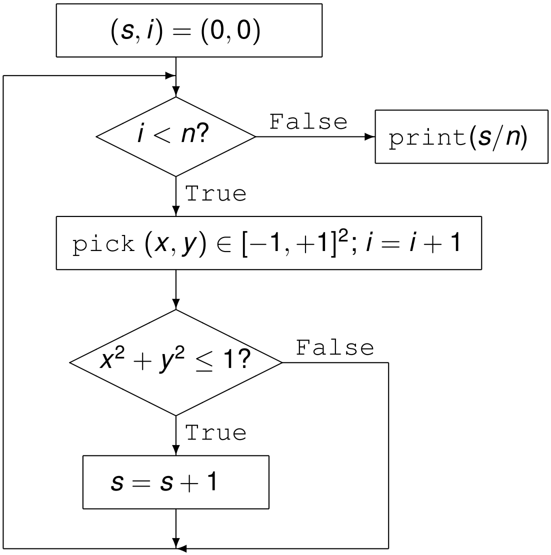
- Expected values are expressed as integrals.
  When many parameters are involved, the integration is high dimensional and only estimation is possible.

- The area of the unit disk is π.



  Generate random uniformly distributed points with coordinates
  $$(x, y) \in [-1, +1] \times [-1, +1].$$
  We count (x, y) as a success if $x^2 + y^2 \leq 1$.

Flowchart for Estimating π



Running Simulations
repeat until: break

1. Simulation
   - Monte Carlo methods
   - volumes and expected values

2. Repeat Until
   - binary expansion
   - break statement

Converting Numbers — from decimal to binary

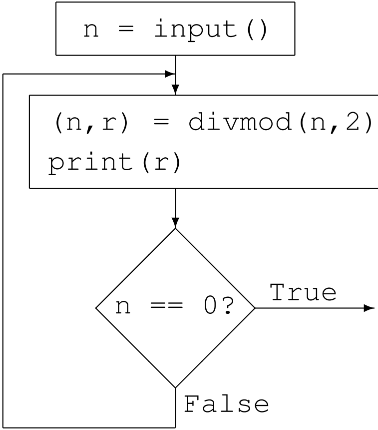
n	$n/2$	$n \bmod 2$	
123	61	1	$123 = 61 \times 2 + 1$
61	30	1	$61 = 30 \times 2 + 1$
30	15	0	$30 = 15 \times 2 + 0$
15	7	1	$15 = 7 \times 2 + 1$
7	3	1	$7 = 3 \times 2 + 1$
3	1	1	$3 = 1 \times 2 + 1$
1	0	1	$1 = 0 \times 2 + 1$

$$
\begin{aligned}
123 &= 1 + 2 \times 61 = 1 + 2 \times (1 + 2 \times 30) \\
&= 1 + 2 \times (1 + 2 \times (0 + 2 \times 15)) \\
&= 1 + 2 \times (1 + 2 \times (0 + 2 \times (1 + 2 \times 7))) \\
&= \ldots = 1111011 = 7B.
\end{aligned}
$$

The *state table* shows the progression of the values of the variables in the loop, each row is one step of the body of the loop.

Flowchart of Binary Expansion

picture of repeat until



Running Simulations
repeat until: break

1. Simulation
   - Monte Carlo methods
   - volumes and expected values
2. Repeat Until
   - binary expansion
   - break statement

The break Statement: repeat until as while true break

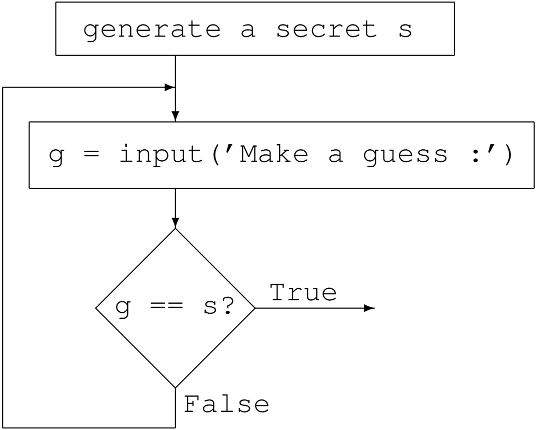
To exit a loop inside the body of a loop,
the statement `break` occurs usually within an `if` statement.

```
while True:
    < body of loop >
    if < condition > :
        break
```

The `while True` starts an infinite loop,
terminated when `< condition >` becomes True.

In a double loop, the `break` leaves only the loop it is in.

Guessing a Secret



Exercises

1. Given a list of numbers between 0 and 100, define the algorithm to assign a letter grade to each number:

   $\geq$ 90: A, $\in [80, 89]$: B, $\in [70, 79]$: C, $\in [60, 69]$: D, else: F.

   Report at the end how many As, Bs, Cs, etc.
   Write the algorithm in words and draw a flowchart.
2. Write a program for the previous exercise.
3. Write a program that generates n numbers uniformly distributed in $[0, 1]$ and counts how many numbers are < 0.5.
4. Use turtle graphics to visualize the Monte Carlo method to estimate π. Represent the unit circle by a circle of radius equal to half of the width of the turtle window. Mark samples inside the disk by green circles of radius equal to 2 pixels, centered at the sample point. Use red circles for the points outside the disk.
5. In the code for guessing of a secret number, add print statements that write `too small` or `too large` when the guess is wrong.