The Birthday Problem through direct calculation and Monte Carlo
In [1]:

```python
using Combinatorics, PyPlot

function sameBirthdayChance(n)
    return 1 - factorial(365, 365-n) / (365^n)
end

grid = 1:50
chances = [Float64(sameBirthdayChance(big(n))) for n in grid]
xlabel("n")
ylabel("prob same birthday")
PyPlot.plot(grid, chances, ".");

using StatsBase
dates = 1:365;

# A random outcome of an experiment: True if there are people with same birthday
#                             False otherwise
function bdEvent(n)
    cnt = counts(sample(dates, n))
    return sum([c > 1 for c in cnt]) > 0  # If there exists at least one greater than 1
end

NN = 10^2

function probEst(n)
    return sum([bdEvent(n) for _ in 1:NN])/NN
end

ests = [probEst(n) for n in grid];

PyPlot.plot(grid, ests, "x");
```