

Additional Exercise

This exercise aims at - simulating the evolution of two cities given some processes that affect that evolution by using Python object oriented design. This exercise is for those with some programming experience to keep them going.

Problem Statement

In this problem, we aim at monitoring the growth of two cities. Each city has a certain population and a certain rate of growth per year: the net percentage growth (or shrinkage) due to births and deaths in the population (assume that such a rate is fixed at the beginning by the system's user). Your task is to devise a computer program that monitors the evolution of the two cities' population over a given period of time (specified, in years, by the user). Every year, each city's population is updated in three ways:

1. because of the normal birth/death process: the population is updated depending on the given rate of growth (so, for instance, if the population is 100 and the rate is 15%, after one year there will be 115 people, after two years there will be 132 people);
2. because of an occasional (rare) catastrophe (that all of a sudden strikes and kills a given number of people): each time such event occurs the population is reduced by a fixed amount. For simplicity assume that there is a one in ten chances of one such event, and the number of people killed each time is a constant value defined in the main application program.
3. because of a "go to the bigger city" attitude. Such a process can be described as follows, with respect to a particular city:

Each year each person in the city decides whether to stay or to leave. With probability 50% the person will stay where (s)he lives. With probability 50% (s)he will decide to relocate to a city selected at random. In such a case the person will relocate depending on the result of the following computation: if pop1 (resp. pop2) is the population of the city where the person lives (resp. the other city), then the person will move if a positive integer number chosen at random in the set containing all positive integers not larger than $pop1 + pop2$, comes up larger than the value of pop1.

The number of people that will move due to the "go to the bigger city" attitude in each city, must be computed first before actually updating the two cities' populations.

The three updates take place in the specified order. Show your results by monitoring two cities A and B over a period of 30 years with initial population sizes of 987,500 and 1000,000; annual growth rate of 7% and 4%; and yearly numbers of possible casualties of 12,000 and 11,000 in case of catastrophe. Discuss the results you have shown in this case.

Further hints

Go in stages! Set up the classes first. Make sure the attributes, the constructors and the data input is done correctly. Then think about the **main** method dynamics. Finally look at each update in turns. Updates 1. and 2. are relatively simple. Aim at having a fully working system implementing just those **BEFORE** moving to the third type of update. The third one is a bit complicated. To generate an event with a “one in ten” chance select a random number in the set $\{0, \dots, 9\}$ and generate the event *every time zero turns up*! This last hint should help you with all probability driven computations.