

An insurance company wants to offer a new 5-year, level-term life insurance policy to recent college graduates. The policy will have a face value (the amount paid in case of death) of \$50,000. Normally, the company charges different premiums depending on the age, gender, tobacco habits and health of the person to be insured*. However, for this simple policy, the company plans to charge a flat \$60/year for every eligible customer, and avoid the underwriting costs normally associated with new policies. It costs the company \$30 (average) for advertising and administration to sell each policy.

In order to estimate the profitability of the new policy, you have been asked to simulate expected policy income (premium payments received) compared to costs (administrative/sale cost plus the occasional death benefit) for an anticipated 100,000 policies.

Background Information:

In each of the 5 policy years, there are 3 possible outcomes.

- The insured may die, and \$50,000 will be paid out (there is a 0.1% chance of death in any one year for persons in this age group)
- The insured may decide to drop the policy, with no payout, and all premiums paid so far are forfeited – (there is a 4% chance of this happening during any year)
- The insured may continue to pay the annual premium and the policy remains in force.

At the end of the fifth year, the insurance terminates, with no further costs to either party.

Hints:

Use a Monte Carlo simulation to solve this problem. Rather than a closed-form solution, Monte Carlo techniques depend on setting up the problem parameters and then inputting a series of random values. The combination of all the random solutions is taken as the problem solution.

Set up the calculations for cash flow each year based on the costs and probabilities given above. Then, the actual cash flow for each policy (each simulation) depends on random numbers which determine which of the possible events actually occurs for that policy holder during that year.

Sum the company's total expected income for all the policies sold, compare this to the cost of selling the policies, and determine how profitable this will be for the company. (You can ignore the cost of money and treat all income as the same, no matter what year it happens.)

To generate random numbers, use the `rand()` function found in `<stdlib.h>`. It does not take any arguments, and it returns an integer between 0 and `RAND_MAX` (a constant already defined in the `stdlib` header). You will also want to initialize, or "seed", the random number function once, at the beginning of your program, by calling `srand(time(NULL))`. Use of the `time()` function requires that you also include the `<time.h>` header.

Test your program for small numbers of random inputs, and hand-verify proper operation before you scale it up to the full 100,000 policies. Run the program several times and observe any differences in the output. If there are differences, what do you think caused them and how should they be controlled.

Collaboration:

Work together as a class (including both 8:30 and 2:30 sections) on any or all aspects of the research and design. Form small teams (1-3 people) to actually code, test and run your program. It is essential to keep track of who did what, and where any useful information was found. Keep a log of the contributions of each member of the team.

What and How to Submit your Solution:

Read the Program Documentation Guidelines, which will be provided. On or before the deadline, hand in (electronically) a file with your program code, and another with sample output data. In class, the day after the deadline, hand in paper copies of the following:

- cover page with program title, date, names of all team members, and the filenames and account name where the files were handed in electronically
- documentation of the subject research you did to prepare for this project (cite sources)
- explanation of steps taken to develop your solution, including any assumptions you made
- program source code, neatly formatted and well-commented
- example output from actual program runs
- explanation of how the program was tested and verified to work correctly
- answer: how profitable is the proposed insurance policy likely to be for the company?
- conclusions: what did you learn? is this a good solution method for this type of problem? What would you change about your approach to this problem if you had to start over?
- collaboration report: acknowledge any non-classmate sources consulted, detail individual contributions of each team member, and identify parts that were completed jointly.

The report should be neatly formatted and typeset (eg: MS Word) with individual sections labeled. Use a single staple in the upper left (no covers). Submit a single report for each team.

Grading Criteria:

- Functionality (20pts): Does it run as specified?
- Features (20pts): Does it have all the specified features (credit for optional features)?
- Logic Design (20pts): Is the design efficient and code well organized?
- Internal Documentation (15pts): Comments, indenting, spacing, variable names?
- Testing (10pts): Was the program thoroughly tested?
- Report Presentation (15pts): Format and completeness of the accompanying report?

Due Date for Electronic Handin is Thursday, Oct 14, at 11:59pm, with paper reports due in class Friday.

* (see <http://www.socialsecurity.gov/OACT/STATS/table4c6.html> for an example actuarial table)