

9.5.2 The Art of Design

It is important to note that spiral development is not an alternative to top-down design. Rather, they are complementary approaches. When designing the prototype, you will still use top-down techniques. In Chapter 12, you will see yet another approach called object-oriented design.

There is no “one true way” of design. The truth is that good design is as much a creative process as a science. Designs can be meticulously analyzed after the fact, but there are no hard-and-fast rules for producing a design. The best software designers seem to employ a variety of techniques. You can learn about techniques by reading books like this one, but books can’t teach how and when to apply them. That you have to learn for yourself through experience. In design, as in almost anything, the key to success is *practice*.

9.6 Chapter Summary

- Computer simulation is a powerful technique for answering questions about real-world processes. Simulation techniques that rely on probabilistic or chance events are known as Monte Carlo simulations. Computers use pseudo-random numbers to perform Monte Carlo simulations.
- Top-down design is a technique for designing complex programs. The basic steps are:
 1. Express an algorithm in terms of smaller problems.
 2. Develop an interface for each of the smaller problems.
 3. Express the algorithm in terms of its interfaces with the smaller problems.
 4. Repeat the process for each of the smaller problems.
- Top-down design was illustrated by the development of a program to simulate the game of racquetball.
- Unit-testing is the process of trying out each component of a larger program independently. Unit-testing and bottom-up implementation are useful in coding complex programs.
- Spiral development is the process of first creating a simple version (prototype) of a complex program and gradually adding features. Prototyping

and spiral development are often useful in conjunction with top-down design.

- Design is a combination of art and science. Practice is the best way to become a better designer.

9.7 Exercises

Review Questions

True/False

1. Computers can generate truly random numbers.
2. The Python `random` function returns a pseudo-random int.
3. Top-down design is also called stepwise refinement.
4. In top-down design, the main algorithm is written in terms of functions that don't yet exist.
5. The main function is at the top of a functional structure chart.
6. A top-down design is best implemented from the top down.
7. Unit-testing is the process of trying out a component of a larger program in isolation.
8. A developer should use either top-down or spiral design, but not both.
9. Reading design books alone will make you a great designer.
10. A simplified version of a program is called a simulation.

Multiple Choice

1. Which expression is true approximately 66% of the time?
a) `random() >= 66` b) `random() < 66`
c) `random() < 0.66` d) `random() >= 0.66`

2. Which of the following is *not* a step in pure top-down design?
 - a) Repeat the process on smaller problems.
 - b) Detail the algorithm in terms of its interfaces with smaller problems.
 - c) Construct a simplified prototype of the system.
 - d) Express the algorithm in terms of smaller problems.
3. A graphical view of the dependencies among components of a design is called a(n)
 - a) flowchart
 - b) prototype
 - c) interface
 - d) structure chart
4. The arrows in a module hierarchy chart depict
 - a) information flow
 - b) control flow
 - c) sticky-note attachment
 - d) one-way streets
5. In top-down design, the subcomponents of the design are
 - a) objects
 - b) loops
 - c) functions
 - d) programs
6. A simulation that uses probabilistic events is called
 - a) Monte Carlo
 - b) pseudo-random
 - c) Monty Python
 - d) chaotic
7. The initial version of a system used in spiral development is called a
 - a) starter kit
 - b) prototype
 - c) mock-up
 - d) beta-version
8. In the racquetball simulation, what data type is returned by the `gameOver` function?
 - a) `bool`
 - b) `int`
 - c) `string`
 - d) `float`
9. How is a percent sign indicated in a string-formatting template?
 - a) `%`
 - b) `\%`
 - c) `%%`
 - d) `\%%`
10. The easiest place in a system structure to start unit-testing is
 - a) the top
 - b) the bottom
 - c) the middle
 - d) the `main` function

Discussion

1. Draw the top levels of a structure chart for a program having the following `main` function:

```
def main():  
    printIntro()  
    length, width = getDimensions()  
    amtNeeded = computeAmount(length,width)  
    printReport(length, width, amtNeeded)
```

2. Write an expression using either `random` or `randrange` to calculate the following:
 - a) A random int in the range 0–10
 - b) A random float in the range -0.5–0.5
 - c) A random number representing the roll of a six-sided die
 - d) A random number representing the *sum* resulting from rolling two six-sided dice
 - e) A random float in the range -10.0–10.0
3. In your own words, describe what factors might lead a designer to choose spiral development over a top-down approach.

Programming Exercises

1. Revise the racquetball simulation so that it computes the results for best of n game matches. First service alternates, so player A serves first in the odd games of the match, and player B serves first in the even games.
2. Revise the racquetball simulation to take shutouts into account. Your updated version should report for both players the number of wins, percentage of wins, number of shutouts, and percentage of wins that are shutouts.
3. Design and implement a simulation of the game of volleyball. Normal volleyball is played like racquetball in that a team can only score points when it is serving. Games are played to 15, but must be won by at least two points.
4. Most sanctioned volleyball is now played using rally scoring. In this system, the team that wins a rally is awarded a point, even if they were not the serving team. Games are played to a score of 25. Design and implement a simulation of volleyball using rally scoring.
5. Design and implement a system that compares regular volleyball games to those using rally scoring. Your program should be able to investigate whether rally scoring magnifies, reduces, or has no effect on the relative advantage enjoyed by the better team.
6. Design and implement a simulation of some other racquet sport (e.g., tennis or table tennis).

- 7.** Craps is a dice game played at many casinos. A player rolls a pair of normal six-sided dice. If the initial roll is 2, 3, or 12, the player loses. If the roll is 7 or 11, the player wins. Any other initial roll causes the player to “roll for point.” That is, the player keeps rolling the dice until either rolling a 7 or re-rolling the value of the initial roll. If the player re-rolls the initial value before rolling a 7, it’s a win. Rolling a 7 first is a loss.

Write a program to simulate multiple games of craps and estimate the probability that the player wins. For example, if the player wins 249 out of 500 games, then the estimated probability of winning is $249/500 = 0.498$.

- 8.** Blackjack (twenty-one) is a casino game played with cards. The goal of the game is to draw cards that total as close to 21 points as possible without going over. All face cards count as 10 points, aces count as 1 or 11, and all other cards count their numeric value.

The game is played against a dealer. The player tries to get closer to 21 (without going over) than the dealer. If the dealer busts (goes over 21), the player automatically wins (provided the player had not already busted). The dealer must always take cards according to a fixed set of rules. The dealer takes cards until he or she achieves a total of at least 17. If the dealer’s hand contains an ace, it will be counted as 11 when that results in a total between 17 and 21 inclusive; otherwise, the ace is counted as 1.

Write a program that simulates multiple games of blackjack and estimates the probability that the dealer will bust. *Hints:* Treat the deck of cards as infinite (casinos use a “shoe” containing many decks). You do not need to keep track of the cards in the hand, just the total so far (treating an ace as 1) and a bool variable `hasAce` that tells whether or not the hand contains an ace. A hand containing an ace should have 10 points added to the total exactly when doing so would produce a stopping total (something between 17 and 21 inclusive).

- 9.** A blackjack dealer always starts with one card showing. It would be useful for a player to know the dealer’s bust probability (see previous problem) for each possible starting value. Write a simulation program that runs multiple hands of blackjack for each possible starting value (ace–10) and estimates the probability that the dealer busts for each starting value.
- 10.** Monte Carlo techniques can be used to estimate the value of pi. Suppose you have a round dartboard that just fits inside of a square cabinet. If

you throw darts randomly, the proportion that hit the dartboard vs. those that hit the cabinet (in the corners not covered by the board) will be determined by the relative area of the dartboard and the cabinet. If n is the total number of darts randomly thrown (that land within the confines of the cabinet), and h is the number that hit the board, it is easy to show that

$$\pi \approx 4\left(\frac{h}{n}\right)$$

Write a program that accepts the “number of darts” as an input and then performs a simulation to estimate π . *Hint:* You can use `2*random() - 1` to generate the x and y coordinates of a random point inside a 2x2 square centered at $(0, 0)$. The point lies inside the inscribed circle if $x^2 + y^2 \leq 1$.

11. Write a program that performs a simulation to estimate the probability of rolling five of a kind in a single roll of five six-sided dice.
12. A *random walk* is a particular kind of probabilistic simulation that models certain statistical systems such as the Brownian motion of molecules. You can think of a one-dimensional random walk in terms of coin flipping. Suppose you are standing on a very long straight sidewalk that extends both in front of and behind you. You flip a coin. If it comes up heads, you take a step forward; tails means to take a step backward.

Suppose you take a random walk of n steps. On average, how many steps away from the starting point will you end up? Write a program to help you investigate this question.

13. Suppose you are doing a random walk (see previous problem) on the blocks of a city street. At each “step” you choose to walk one block (at random) either forward, backward, left or right. In n steps, how far do you expect to be from your starting point? Write a program to help answer this question.
14. Write a graphical program to trace a random walk (see previous two problems) in two dimensions. In this simulation you should allow the step to be taken in *any* direction. You can generate a random direction as an angle off of the x axis.

```
angle = random() * 2 * math.pi
```

The new x and y positions are then given by these formulas:

$$x = x + \cos(\text{angle})$$

$$y = y + \sin(\text{angle})$$

The program should take the number of steps as an input. Start your walker at the center of a 100x100 grid and draw a line that traces the walk as it progresses.

15. (Advanced) Here is a puzzle problem that can be solved with either some fancy analytic geometry (calculus) or a (relatively) simple simulation.

Suppose you are located at the exact center of a cube. If you could look all around you in every direction, each wall of the cube would occupy $\frac{1}{6}$ of your field of vision. Suppose you move toward one of the walls so that you are now halfway between it and the center of the cube. What fraction of your field of vision is now taken up by the closest wall? *Hint:* Use a Monte Carlo simulation that repeatedly “looks” in a random direction and counts how many times it sees the wall.