

Homework 4

10 试题

1
point

1.

"Invention consists in avoiding the constructing of useless contraptions and in constructing the useful combinations which are in infinite minority",

Henri Poincare

Combinatorics

In this week's material, we will create enumerations, permutations and combinations of items from a set of outcomes. We will then consider sequences of repeated trials that are modeled by these objects. These problems will form our preparation for this week's mini-project on Yahtzee.

Enumeration

Given the set of outcomes corresponding to a coin flip, $\{Heads, Tails\}$, how many sequences of outcomes of length five (repetition allowed) are possible?

32

1
point

2.

Probability for sequences of trials

Consider a sequence of trials in which a fair four-sided die (with faces numbered 1-4) is rolled twice. What is the expected value of the product of the two die rolls? Enter the answer as a floating point number below.

6.25

1
point

3.

Given a trial in which a decimal digit is selected from the list ["0", "1", "2", "3", "4", "5", "6", "7", "8", "9"] with equal probability 0.1, consider a five-digit string created by a sequence of such trials (leading zeros and repeated digits are allowed). What is the probability that this five-digit string consists of five consecutive digits in either ascending or descending order (e.g; "34567" or "43210") ?

Enter your answer as a floating point number with at least four significant digits of precision.

0.0001200

1
point

4.

Permutations

Consider a trial in which five digit strings are formed as permutations of the digits

`["0", "1", "2", "3", "4", "5", "6", "7", "8", "9"]`

. (In this case, repetition of digits is not allowed.) If the probability of each permutation is the same, what is the probability that this five digits string consists of consecutive digits in either ascending or descending order (e.g; `"34567"` or `"43210"`) ?

Enter your answer as a floating point number with at least four significant digits of precision.

0.0003968

1

point

5.

In this week's lectures, we discussed an iterative approach to generating all sequences of outcomes where repeated outcomes were allowed. Starting from this program template, implement a function

`gen_permutations(outcomes, num_trials)` that takes a list of outcomes and a number of trials and returns a set of all possible permutations of length **`num_trials`** (encoded as tuples) from this set of outcomes.

Hint: **`gen_permutations`** can be built from **`gen_all_sequences`** by adding a single **`if`** statement that prevents repeated outcomes. When you believe that your code works correctly, select the answer printed at the bottom of the console.

☒ ('b', 'e', 'c', 'd')

☐ ('f', 'a', 'b', 'c')

☐ ('e', 'b', 'd', 'c')

☐ ('a', 'f', 'b', 'e')

1
point

6.

Subsets

A set S is a *subset* of another set T (mathematically denoted as $S \subseteq T$) if every element x in S (mathematically denoted as $x \in S$) is also a member of T . Which of the following sets are subsets of the set $\{1, 2\}$?

☒ $\{2\}$

☐ $\{3, 4\}$

☐ $\{1, 2, 3, 4\}$

☒ $\{\}$

☒ $\{1, 2\}$

1
point

7.

If the set T has n members, how many distinct sets S are subsets of T ? You may want to figure out the answer for a few specific values of n first. Enter the answer below as a math expression in n .

预览

$$2^n$$

2^n

1
point



8.
Combinations

Given a standard 52 card deck of playing cards, what is the probability of being dealt a five card hand where all five cards are of the same suit?

Hint: Use the formula for combinations to compute the number of possible five card hands when the choice of cards is restricted to a single suit versus when the choice of cards is unrestricted.

Compute your answer in Python using `math.factorial` and enter the answer below as a floating point number with at least four significant digits of precision.

0.00002001

1
point

9.
Pascal's triangle is a triangular array of numbers in which the entry on one row of the triangle corresponds to the sum of the two entries directly above the entry. This program prints out the first few rows of Pascal's triangle.

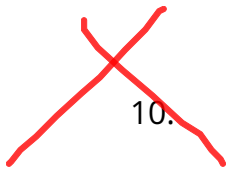
Enter a math expression in m and n using factorial (!) that represents the value of the n th entry of the m th row of Pascal's triangle. (Both the row numbers and entry numbers are indexed starting at zero.)

预览

$$\frac{m!}{n!(m-n)!}$$

factorial(m)/(factorial(n)*factorial(m-n))

1
point



For the final question of this week's homework, your task is to create a list of test cases for the method `plan_moves()` from the `SolitaireMancala` class in this Practice Activity. The purpose in including problems of this type is to focus your attention on the process of creating test cases on your own and not relying entirely on OwlTest. To assess the quality of your test cases, we have created a series of OwlTests that automatically assesses how well your test cases detect erroneous programs written by your peers.

For this question, your task is to create a list of test cases for the method `plan_moves()` from `Solitaire Mancala`. Each test case in this list should itself be a list of integers of length at least 1 and at most 10, whose entries are integers in the range 0 to 10 (inclusive). Each of these lists corresponds to a configuration of the `Solitaire Mancala` board prior to calling `plan_moves()`. OwlTest will take each test case in your supplied list and use the `set_board()` method to set the board to the configuration corresponding to the test case. OwlTest will then call `plan_moves()` on each configuration and compare the returned list of moves versus that generated by our reference implementation of the `SolitaireMancala` class. For this question, your supplied test cases should be restricted to configurations in which the store is empty prior to the call to `plan_moves()` (i.e; the first integer in the list for the test case is zero).

To complete this problem, visit this OwlTest page and follow the directions for creating and submitting a list of test cases. Once OwlTest has successfully assessed your test cases, you will see the message

TEST CASES successfully assessed. . Following this message is a seven-digit number that you should enter in the form below. For this task, **please ignore the fact that this message appears under the red Unit Test Failures tab.**

This OwlTest automatically assesses how effective your list of test cases is in detecting erroneous programs from a suite of implementations of the `SolitaireMancala` class compiled from previous sessions of PoC. If you do not catch all of the erroneous programs, OwlTest outputs an example of one erroneous program that passes all of your submitted test cases. You must catch all of the erroneous programs to get this question correct. The homework feedback will tell you what percent of erroneous programs you were able to catch.

Observe that, since `plan_moves()` is part of the `SolitaireMancala` class, an error in another method in this class may cause `plan_moves()` to also produce incorrect results.

在此输入您的回答

1 试题 未回答

提交测试

