Quiz 6: Recurrent Sequences

Question 1: Modifying Sudoku Rules.
We have a 9×9 table; each cell contains one number from $A_9 = \{1, 2, \ldots, 9\}$. Number $a_{i,j}$ describes the number written on Line i and Column j in the table (and $i, j \in A_9$).
Identify the formula describing that the sum of all numbers in one diagonal of the table 9×9 is equal to the sum of all numbers on the other diagonal of the table.

(A) $\sum_{r=0}^{8} a_{r+1,r+1} = \sum_{r=0}^{8} a_{9-r,r+1}$

(B) $\sum_{r=0}^{8} a_{r,r} = \sum_{r=0}^{8} a_{9-r,r}$

(C) $\sum_{r=0}^{8} a_{r+1,r+1} = \sum_{r=0}^{8} a_{10-r,r+1}$

(D) $\sum_{r=0}^{8} a_{r,r} = \sum_{r=0}^{8} a_{10-r,r}$

Question 2: Computing sums.
Find the values of these sum in a Sudoku table (it is known that each number from $A_9 = \{1, 2, \ldots, 9\}$ is written exactly once in all rows, in all columns and also in all 3×3 blocks).

$$\sum_{r=1}^{9} \sum_{s=1}^{9} a_{r,s}$$

Write in the number as your answer.

Question 3: Long set operations. Denote $A_1 = \{1\}$, $A_2 = \{1, 2\}$, etc. In general, $A_k = \{1, 2, \ldots, k\}$.
By $A \oplus B = (A - B) \cup (B - A)$ we denote the symmetric difference: All elements that belong to just one of the sets A, B (but not the other one). Consider this set:

$$S = \bigoplus_{j=1}^{100} A_{2j-1}.$$

Write a comma-separated list of the 10 smallest elements of S in increasing order.

Question 4: Using recurrent formula. Find the first 6 members of this infinite sequence $(C_0, C_1, C_2, \ldots)$:

$$\begin{cases} C_0 = 1, \\ C_{n+1} = \sum_{i=0}^{n} (C_i \cdot C_{n-i}). \end{cases}$$

In your answer write comma-separated values:
$C_0, C_1, C_2, C_3, C_4, C_5$.

Question 5: Reccurent sequence. A sequence of real numbers $f : \mathbb{N} \to \mathbb{R}$ satisfies the following properties:
(A) $f(k+2) = 2f(k+1) + f(k)$ for all integers $k \geq 2$.
(B) $f(n)$ is a geometric progression.
Find some quotient of this geometric progression (if there are several possibilities, pick any of them). Round it to the nearest thousandth, i.e. specify the first three digits after the decimal point).

Question 6: Finding a limit. Define the following sequence:

$$\begin{cases} x_0 = 1, \\ x_{n+1} = \frac{1}{3}\left(2x_n + \frac{7}{x_n^2}\right), \text{ if } n \geq 0 \end{cases}$$

Assume that there exists limit $L = \lim_{n\to\infty} x_n$.
Find that limit L and round it to the nearest thousandth.

Question 7: Taylor series There is a formula known from calculus (practically used to compute $y = \sin x$) for each $x \in \mathbb{R}$.

$$\sin x = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot x^{2n+1}}{(2n+1)!} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \ldots$$

Use Python or Scala to add the first 20 terms of this infinite sum to compute $\sin 1080^\circ$. Round the answer to 15 digits (this usually happens by itself: double precision numbers are output in this manner).

Answers

Question 1. Answer: `A`
All other variants refer to a_{ij}, where i or j are outside the interval $[1;9]$, so they are not defined. Answer (A), on the other hand, refers to the elements that lay on both diagonals.

Question 2. Answer: `405`
This summation is the total of all elements in the Sudoku table. Each row contains all numbers from 1 to 9. So they add up to $45 = \frac{9\cdot(9+1)}{2}$. This has to be multiplied by 9, since there are 9 rows.

Question 3. Answer:
`2,3,6,7,10,11,14,15,18,19`
The smallest number $1 \notin S$, because it belongs to all 100 sets (so it cancels out, when we compute the long symmetric sum).
Numbers 2 and 3 belong to exactly 99 sets, therefore they are included.
Numbers 4 and 5 belong to exactly 98 sets, so they are not included, and so on.

Question 4. Answer: `1,1,2,5,14,42`
We can simply plug into the formulas:

$$\begin{cases} C_1 = C_0 \cdot C_0 = 1 \cdot 1 = 1 \\ C_2 = C_0 \cdot C_1 + C_1 \cdot C_0 = 1 \cdot 1 + 1 \cdot 1 = 2 \\ C_3 = C_0 \cdot C_2 + C_1 \cdot C_1 + C_2 \cdot C_0 = 5 \\ \ldots \end{cases}$$

These are also known as *Catalan numbers*. For example C_3 shows, in how many "essentially different" ways you can put parentheses in an expression with four letters and three binary operators:

$$(((a \circ b) \circ c) \circ d);\ ((a \circ (b \circ c)) \circ d);$$

$$((a \circ b) \circ (c \circ d));\ (a \circ ((b \circ c) \circ d));$$

$$(a \circ (b \circ (c \circ d)))$$

Here we assume that the $\circ$ operation is neither associative nor commutative. Certainly, we could insert even more parentheses; but these 5 ways differ by the order of execution of these three operations.

Question 5. Answer: `2.414` **or** `-0.414`
If we substitute the geometric progression $f(n) = f(0) \cdot q^n$ into the equation $f(k+2) = 2f(k+1) + f(k)$:

$$f(0) \cdot q^{k+2} = 2f(0) \cdot q^{k+1} + f(0) \cdot q^k.$$

Now there are three possibilities.
Case 1. If $f(0) = 0$, then the sequence $0, 0, \ldots$ is a (degenerated) version of a geometric series. Any number can be its quotient; so it is not interesting to solve this.
Case 2. If $f(0) \neq 0$, we could still take $q = 0$, but this is also a degenerated geometric series of all zeroes (except the first member $f(0)$). This is not interesting either.
Case 3. If $f(0)$ and q are both nonzero, we can cancel them in the above equation and get this: $q^2 = 2q + 1$. The two roots are

$$q_{1,2} = \frac{2 \pm \sqrt{4+4}}{2} = 1 \pm \sqrt{2}.$$

Either answer $1 - \sqrt{2}$ or $1 + \sqrt{2}$ is valid.

Question 6. If the limit L exists, then both x_n and x_{n+1} both go to that limit, and we must have the identity:

$$L = \frac{1}{3}\left(2L + \frac{7}{L^2}\right).$$

When we express L from that equation, we get $L = \sqrt[3]{7}$. In fact, the sequence does converge to the cubic root:

```
x(0) = 1.000000000000000
x(1) = 3.000000000000000
x(2) = 2.259259259259259
x(3) = 1.963308018221572
x(4) = 1.914212754165601
x(5) = 1.912932040596942
x(6) = 1.912931182772774
x(7) = 1.912931182772389
x(8) = 1.912931182772389
```

Question 7. Answer:
`-482.8813747415584` (or similar)
In theory $\sin 1080° = 0$ (and Taylor series should converge to 0), but this nonsense result happens because of huge rounding errors.

```
object TaylorSeries {

  def factorial(n:Int):Double = {
    n match {
      case 0 => 1.0
      case n => n*factorial(n-1)
    }
  }

  def sin(x:Double,nTerms:Int):Double = {
    val terms =
      for (k <- List.range(0,nTerms))
        yield { math.pow(-1,k)*
        math.pow(x,2*k+1)/factorial(2*k+1) }
    terms.foldLeft(0.0)((a,b) => a+b)
  }

  def main(args:Array[String]): Unit = {
    val nTerms = 20
    val x = Math.PI*6 // (1080/180)*PI = 6PI
    println(sin(x,nTerms))
  }
}
```