

Задачи 3.2. Функции обработки матриц.

- 1) Описать функцию RemoveRowCol, удаляющую из матрицы A размера $M \times N$ строку и столбец, которые содержат элемент $A_{K,L}$ (предполагается, что $M > 1$ и $N > 1$; если $K > M$ или $L > N$, то матрица не изменяется).
- 2) Описать функцию Split, формирующую по заданному списку A два новых списка B и C ; при этом список B содержит все чётные числа из списка A , а список C — все нечётные числа (в том же порядке).
- 3) Описать функцию SwapRow, осуществляющую перемену местами строк вещественной матрицы A размера $M \times N$ с номерами K_1 и K_2 .
- 4) Описать функцию Transp, выполняющую транспонирование (то есть зеркальное отражение относительно главной диагонали) квадратной матрицы A порядка M .
- 5) Описать функцию MtrProd, вычисляющую произведение двух матриц.
- 6) Описать функцию SumMtr, вычисляющую сумму двух матриц.
- 7) Дана квадратная матрица. Вычислить сумму элементов главной или побочной диагонали в зависимости от выбора пользователя. Сумма элементов любой диагонали должна вычисляться в одной и той же функции.
- 8) Описать функцию RemoveRows, удаляющую из матрицы A размера $M \times N$ строки с номерами от K_1 до K_2 включительно (предполагается, что $1 < K_1 \leq K_2$). Если $K_1 < M$, то матрица не изменяется; если $K_2 > M$, то удаляются строки матрицы с номерами от K_1 до M .
- 9) Описать функцию InvertStr, возвращающую инвертированную подстроку строки S , содержащую в обратном порядке N символов строки S , начиная с её K -го символа. Если K превосходит длину строки S , то возвращается пустая строка; если длина строки меньше $K + N$, то инвертируются все символы строки, начиная с её K -го символа.
- 10) Описать функцию SwapCol, осуществляющую перемену местами столбцов матрицы A размера $M \times N$ с номерами K_1 и K_2 .
- 11) Описать функцию CompressStr, выполняющую сжатие строки S по следующему правилу: каждая подстрока строки S , состоящая из более чем четырёх одинаковых символов C , заменяется текстом вида « $C\{K\}$ », где K — количество символов C (предполагается, что строка S не содержит фигурных скобок « $\{$ » и « $\}$ »). Например, для строки $S = bbbccccc$ функция вернет строку $bbbc\{5\}e$.
- 12) Описать функцию IsIdent, проверяющую, является ли строка S допустимым идентификатором, то есть непустой строкой, которая содержит только латинские буквы, цифры и символ подчеркивания « $_$ » и не начинается с цифры.

- 13) Описать функцию Chessboard, формирующую по целым положительным числам M и N матрицу A размера $M \times N$, которая содержит числа 0 и 1, расположенные в «шахматном» порядке, причём $A_1 = 0$.
- 14) Описать функцию Bell, меняющую порядок элементов списка L на следующий: наименьший элемент списка располагается на первом месте, наименьший из оставшихся элементов — на последнем, следующий по величине располагается на втором месте, следующий — на предпоследнем и т. д. (в результате график значений элементов будет напоминать колокол).
- 15) Описать функцию RemoveCols, удаляющую из матрицы A размера $M \times N$ столбцы с номерами от K_1 до K_2 включительно (предполагается, что $1 < K_1 \leq K_2$). Если $K_1 > N$, то матрица не изменяется; если $K_2 > N$, то удаляются столбцы матрицы с номерами от K_1 до N .
- 16) Описать функцию SplitEvenOdd, которая разделяет список чисел на два списка: список четных чисел и список нечетных чисел.
- 17) Описать функцию ReverseTuples, которая принимает список кортежей и возвращает новый список, где каждый кортеж перевернут в обратном порядке.
- 18) Описать функцию ReplaceDuplicates, которая заменяет все дублирующиеся элементы в списке на заданное значение.
- 19) Описать функцию SquaredSum, которая вычисляет сумму квадратов элементов списка.
- 20) Описать функцию UniqueIntersection, которая принимает два списка и возвращает список уникальных общих элементов обоих списков.
- 21) Описать функцию SplitByCondition, которая разделяет список на два списка: один содержит элементы, удовлетворяющие заданному условию, а второй содержит остальные элементы.
- 22) Описать функцию RemoveDuplicatesList, которая удаляет все повторяющиеся элементы из списка.
- 23) Описать функцию TransposeMatrix, которая принимает матрицу в виде списка списков и возвращает ее транспонированную версию.
- 24) Описать функцию MergeDicts, которая принимает произвольное количество словарей и объединяет их в один словарь.
- 25) Описать функцию CountSublists, которая принимает список и возвращает количество вложенных списков внутри него.
- 26) Описать функцию SwapElements, которая принимает список и индексы двух элементов и меняет их местами.

- 27) Описать функцию `CountOccurrences`, которая подсчитывает количество вхождений каждого элемента в списке и возвращает словарь, где ключи — элементы списка, а значения — количество их вхождений.
- 28) Описать функцию `SortNestedList`, которая принимает список вложенных списков и сортирует его по возрастанию суммы элементов в каждом вложенном списке.
- 29) Описать функцию `FlattenList`, которая принимает вложенный список и возвращает одноуровневый список, содержащий все элементы из вложенных списков.
- 30) Описать функцию `RemoveEmptyStrings`, которая удаляет пустые строки из списка строк.