Міністерство освіти і науки України

Національний університет «Запорізька політехніка»

кафедра програмних засобів

ЗВІТ

з лабораторної роботи №5

з дисципліни «**Спортивне програмування**» на тему:

**«**ДИНАМІЧНЕ ПРОГРАМУВАННЯ»

Виконав:

ст. гр. КНТ-113сп Артем КЛЕЩОВ

Прийняв:

ст. викл.: Сергій ЛЕОЩЕНКО

2024

# Мета роботи

1.1 Вивчити основні можливості та принципи роботи із підходом динамічного програмування.

# Завдання до лабораторної роботи

2.2.3 Потрібно обчислити кількість N-значних чисел в системі числення з основою K, таких що їх запис не містить двох поспіль йдуть нулів.

Вхідні дані у вхідному файлі INPUT.Txt записані два натуральних числа N і K в десятковій системі числення (2 ≤ K ≤ 10; 2 ≤ N; 4 ≤ N+K ≤ 18).

Вихідні дані: у вихідний файл OUTPUT.TXT необхідно вивести ціле число в десяткового запису – відповідь на задачу.

2.2.7 Потрібно обчислити кількість N-значних щасливих квитків. Нагадаємо, що квиток називається щасливим, якщо сума першої половини його цифр дорівнює сумі іншої його половини. Наприклад, квиток 064109 щасливий, тому що. 0+6+4=1+0+9.

Вхідні дані: у єдиному рядку вхідного файлу INPUT.TXT записано натуральне парне число N (N ≤ 100) – кількість цифр у квитку.

Вихідні дані: В єдиний рядок вихідного файлу OUTPUT.TXT потрібно вивести одне ціле число – кількість N-значних щасливих квитків.

# 3. Текст розробленої програми

Задача 3:

function countNumbers(*N*, *K*) {

    if (!(*N* >= 2 && *K* >= 2 && *K* <= 10 && (*N* + *K*) >= 4 && (*N* + *K*) <= 18)) {

        return 0;

    }

    let count = 0;

    function helper(*digitsLeft*, *prevDigit*) {

        if (*digitsLeft* === 0) {

            return 1;

        }

        let total = 0;

        for (let digit = 0; digit < *K*; digit++) {

            if (digit !== 0 || *prevDigit* !== 0) {

                total += helper(*digitsLeft* - 1, digit);

            }

        }

        return total;

    }

    for (let startDigit = 1; startDigit < *K*; startDigit++) {

        count += helper(*N* - 1, startDigit);

    }

    return count;

}

const fs = require('fs');

fs.readFile('INPUT.txt', 'utf8', function(*err*, *data*) {

  if (*err*) {

      return console.log(*err*);

  }

  const [N, K] = *data*.trim().split(' ').map(Number);

  const result = countNumbers(N, K);

  fs.writeFile('OUTPUT.txt', result.toString(), function(*err*) {

      if (*err*) {

          return console.log(*err*);

      }

      console.log('Output has been saved to OUTPUT.txt');

  });

});

Задача 7:

const fs = require('fs');

function countLuckyTickets(*N*) {

    if (*N* % 2 !== 0) return 0;

    const dp = new Array(*N* / 2 + 1).fill(null).map(() => new Array(*N* \* 9 / 2 + 1).fill(0));

    dp[0][0] = 1;

    for (let i = 1; i <= *N* / 2; i++) {

        for (let j = 0; j <= i \* 9; j++) {

            for (let k = 0; k <= 9; k++) {

                if (j - k >= 0) {

                    dp[i][j] += dp[i - 1][j - k];

                }

            }

        }

    }

    let total = 0;

    for (let j = 0; j <= *N* \* 9 / 2; j++) {

        total += dp[*N* / 2][j] \* dp[*N* / 2][j];

    }

    return total;

}

fs.readFile('INPUT.TXT', 'utf8', (*err*, *data*) => {

    if (*err*) {

        console.error(*err*);

        return;

    }

    const N = parseInt(*data*.trim());

    const result = countLuckyTickets(N);

    fs.writeFile('OUTPUT.TXT', result.toString(), (*err*) => {

        if (*err*) {

            console.error(*err*);

            return;

        }

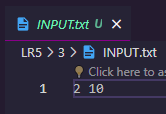
        console.log('Output has been saved to OUTPUT.txt');

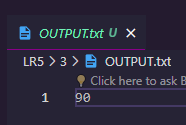
    });

});

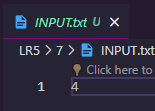
# 4. Копії екрану, що відображають результати виконання лабораторної роботи.

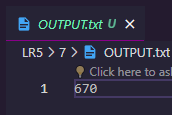
Задача 3:





Задача 7:





# 5. Висновки

Під час виконання лабораторної роботи, я вивчив основні можливості та принципи роботи із підходом динамічного програмування.