



МИНОБРНАУКИ РОССИИ
*Федеральное государственное бюджетное образовательное учреждение высшего
образования*
«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Отчет по выполнению практического задания №8.2

Тема:

Реализация алгоритмов на основе сокращения числа
переборов

Дисциплина: «Структуры и алгоритмы обработки данных»

Выполнил студент: Васильев Б.А.

Группа: ИКБО-20-23

Москва 2024

СОДЕРЖАНИЕ

ЦЕЛЬ РАБОТЫ	3
ХОД РАБОТЫ	3
Формулировка задачи	3
Описание подхода к решению	3
Код программы.....	5
Результаты тестирования	6
ВЫВОД.....	7
СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ	7

ЦЕЛЬ РАБОТЫ

Освоить приёмы реализации алгоритмов с использованием методов сокращения числа переборов для эффективного решения задач.

ХОД РАБОТЫ

Формулировка задачи

1. Разработать алгоритм решения задачи с применением метода, указанного в варианте и реализовать программу.
2. Оценить количество переборов при решении задачи стратегией «в лоб» - грубой силы. Сравнить с числом переборов при применении метода.
3. Оформить отчет в соответствии с требованиями документирования разработки ПО: Постановка задачи, Описание алгоритмов и подхода к решению, Код, результаты тестирования, Вывод.

№_	Задача	Метод
1	Посчитать число последовательностей нулей и единиц длины n , в которых не встречаются две идущие подряд единицы.	Динамическое программирование

Рисунок 1 – Индивидуальный вариант задачи

Описание подхода к решению

При $n < 32$ полный перебор потребует нескольких секунд, а при $n = 64$ полный перебор не осуществим в принципе. Для решения задачи методом динамического программирования сведем исходную задачу к подзадачам.

При $n = 0$, $n = 1$ ответ очевиден. Допустим, что мы уже нашли K_{n-1} , K_{n-2} — число таких последовательностей длины $n - 1$ и $n - 2$.

Посмотрим, какой может быть последовательность длины n . Если последний ее символ равен 0, то первые $n - 1$ — любая правильная последовательность длины $n - 1$ (не важно, заканчивается она нулем или единицей — следом идет 0). Таких последовательностей всего K_{n-1} .

Если последний символ равен 1, то предпоследний символ обязательно должен быть равен 0 (иначе будет две единицы подряд), а первые $n - 2$ символа — любая правильная последовательность длины $n - 2$, число таких последовательностей равно K_{n-2} .

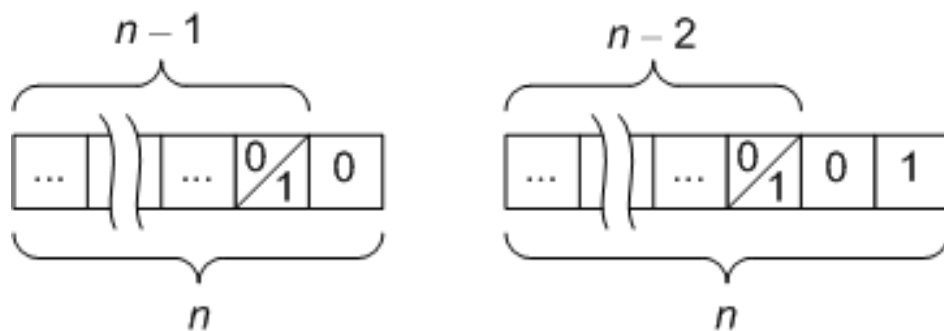


Рисунок 2 – удовлетворяющие условию последовательности

Таким образом, $K_0 = 1$, $K_1 = 2$, $K_n = K_{n-1} + K_{n-2}$ при $n > 2$. То есть данная задача фактически сводится к нахождению чисел Фибоначчи.

Код программы

Реализуем код приложения на языке программирования C++ (рис. 3).

```
#include <iostream>
#include <vector>

using namespace std;

int countSequences(int n) {
    if (n == 0) return 1;
    if (n == 1) return 2;

    int dp[n+1];
    dp[0] = 1;
    dp[1] = 2;

    for (int i = 2; i <= n; ++i) {
        dp[i] = dp[i - 1] + dp[i - 2];
    }

    return dp[n];
}

int main() {
    setlocale(LC_ALL, "ru_RU.UTF-8");
    int n;
    cout << "Введите длину последовательности n: ";
    cin >> n;

    int result = countSequences(n);
    cout << "Число последовательностей длины " << n << " без подряд идущих единиц: " << result << endl;

    cout << endl;
    cout << "Число переборов при динамическом программировании: " << n << endl;
    // Оценка числа переборов при стратегии "в лоб"
    cout << "Число переборов при стратегии \"в лоб\": " << (1 << n) << endl;

    return 0;
}
```

Рисунок 3 – код программы

Результаты тестирования

Выполним тестирование программы на различных значениях n . (рис. 4-7)

```
Введите длину последовательности n: 1
Число последовательностей длины 1 без подряд идущих единиц: 2

Число переборов при динамическом программировании: 2
Число переборов при стратегии "в лоб": 2
```

Рисунок 7 – Пример текстового файла

```
Введите длину последовательности n: 2
Число последовательностей длины 2 без подряд идущих единиц: 3

Число переборов при динамическом программировании: 3
Число переборов при стратегии "в лоб": 4
```

Рисунок 7 – Пример текстового файла

```
Введите длину последовательности n: 5
Число последовательностей длины 5 без подряд идущих единиц: 13

Число переборов при динамическом программировании: 6
Число переборов при стратегии "в лоб": 32
```

Рисунок 7 – Пример текстового файла

```
Введите длину последовательности n: 32
Число последовательностей длины 32 без подряд идущих единиц: 5702887

Число переборов при динамическом программировании: 33
Число переборов при стратегии "в лоб": 4.29497e+09
```

Рисунок 7 – Пример текстового файла

Тестирование показало, что программа работает корректно.

ВЫВОД

В результате выполнения работы были освоены приёмы реализации алгоритмов с использованием методов сокращения числа переборов для эффективного решения задач.

СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ

1. Рысин, М. Л. Введение в структуры и алгоритмы обработки данных : учебное пособие / М. Л. Рысин, М. В. Сартаков, М. Б. Туманова. — Москва : РТУ МИРЭА, 2022 — Часть 2 : Поиск в тексте. Нелинейные структуры данных. Кодирование информации. Алгоритмические стратегии — 2022. — 111 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/310826> (дата обращения: 02.11.2024).
2. Документация по языку C++ [Электронный ресурс]. URL: <https://docs.microsoft.com/ruru/cpp/cpp/> (дата обращения 02.11.2024).