dz_1

1. Постановка задачи (условие задачи)

Условие задания по варианту.

> В текстовом файле содержался целые числа. Найти сумму цифр каждого числа и если она четная, то перевести его в двоичную систему счисления, а если нечетная, то преобразовать число в обратном порядке следования цифр. Результат вывести на экран.

Допущения

- В файле только целые числа которые записаны правильно
- Числа в файле не больше 100
- Количество чисел файла не указывается
- Числа разделяются одним или несколькими пробелами и символами перевода на другую строку
- В числах записанных в двоичной системе первый бит означает за знак

Примеры работы программы("Черный ящик")

Исх. файл	Вывод
Несуществующий файл	Сообщение "Файл не найден"
Пустой файл	Сообщение "Пустой файл"
В файле больше 100 чисел	Сообщение "В файле больше 100 чисел. Превышен предел."
a b c r y 7	Сообщение "Файл записан некорректно. Содержит буквы и символы."
1.23 6.78 9.45 10.0	Сообщение "Файл записан некорректно. Содержит вещественные числа."
-231 121 1234	111100111 01111001 010011010010
120 -313 975	12 -313 579
234 222 -234	432 01101110 -432
-342 -234 -121	-243 -432 11111001
0	0
121 1234 120 313 975 450 0 1	01111001 010011010010 12 313 579 54 0 1

2 - 3. Декомпозиция и Алгоритмы

Диаграммы IDEF0

Спецификации модулей (выделенных подзадач)

- Спецификация модуля A1

Имя	ReadArray
Функция	Считывает в массив числа расположенные в файле
Список параметров	Файловая переменная массив количество элементов в массиве
Входные данные	Файловая переменная
Выходные данные	Массив количество элементов в массиве
Внешние эффекты	-

Тесты	Результат
Пустой	Массив пустой количество элементов равно 0
123 567 -123	123 567 -123 123 6 7 8 1 2, количество элементов 9
123 6 7 8 1 2	123 567 -123 123 6 7 8 1 2, количество элементов 9
1 2 3 4 5 6	1 2 3 4 5 6, количество элементов 6

- Псевдокод модуля A1

```
ReadArray
Вход:   f - файловая переменная
Выход:  a - массив целых чисел,
        n - количество элементов в массиве
нач
пока не конец файла делать
    прочитать элемент массива a[n]
    n=n+1
все пока
```

- Спецификация модуля A3

Имя	SumOfFigure
Функция	Считает сумм цифр в числе
Список параметров	Число, сумма цифр в числе
Входные данные	Число
Выходные данные	Сумма цифр в числе
Внешние эффекты	-

Тесты	Результат
123	6
-456	15
890	17

- Псевдокод модуля A3

```
Вход:   number - число
Выход:  sum - сумма цифр в числе

sum = 0
number = abs(number)
пока number больше нуля делать
    sum = sum + (number % 10)
    number = number / 10
все пока
```

- Спецификация модулей A4

Имя	CheckSum
Функция	Проверяет четность числа
Список параметров	Число, признак
Входные данные	Число
Выходные данные	Признак
Внешние эффекты	-

Тесты	Результат(1 - True, 0 - False)
6	1
15	0
102	1

- Псевдокод модуля A4

```
Вход:   number - число
Выход:  response - признак

response = 0
number = abs(number)
если number % 2 равно 0 то
    response = 1
иначе
    response = 0
все если
```

- Спецификация модулей A5

Имя	ConvertingToBinaryCod
Функция	Преобразует числа в двоичный код
Список параметров	Число, массив двоичного кода, размер массива
Входные данные	Число
Выходные данные	массив двоичного кода, размер массива
Внешние эффекты	-

Тесты	Элементы массива, размерность
121	01111001, 8
5	0101, 4
-46	1101110, 7
23256	010010101101000, 16
-23256	110101010110100, 16

- Псевдокод модуля A5

```
Вход:   number - число
Выход:  binarycod - массив двоичного кода,
        nb - размер массива
nb = 0
если number меньше 0 то
    sign = 1
иначе
    sign = 0
все если
number = abs(number)
пока number больше 0 делать
    если number % 2 равно 0 то
        b[nb] = 0
    иначе
        b[nb] = 1
    все если
    number = number / 2
    nb = nb + 1
все пока
b[nb + 1] = sign

i = nb
пока i больше и равно nb / 2 то
    b[i] = b [nb - i - 1]
    b[nb - i - 1] = b [i]
    i = i - 1
все пока
Примечание:
            sign - первый бит, определяющий знак двоичного кода
            i - подсчет итераций для числа начиная с конца массива
```

- Спецификация модулей A6

Имя	ReverseNumber
Функция	Преобразует числа в число в обратном порядке следования
Список параметров	Число, число в обратном порядке следования
Входные данные	Число
Выходные данные	Число в обратном порядке следования
Внешние эффекты	-

Тесты	Результат
132	221
247	742
-47	-74
2345787	7875432
-662377	-773289

- Псевдокод модуля A6

```
Вход:   number - число
Выход:  reverse - число в обратном порядке следования

reverse = 0
пока number больше 0 делать
    reverse = reverse * 10 + number % 10
    number = number / 10
все пока
```

- Псевдокод основной программы

```
Вход:  имя файла
Выход: преобразованные числа

Открыть файл my_file с указанным именем
если файл существует и он не пустой то
    ReadArray(my_file, arr, n)
    пока не конец массива a то
        SumOfFigure(number, sum)
        если CheckSum(sum, response) равно 1 то
            ConvertingToBinaryCod(number, binarycod)
            i = 0
            пока i < nb делать
                напечатать binarycod[i]
                i = i + 1
            все пока
        иначе
            ReverseNumber(number, reverse)
            напечатать reverse
        все если
    все пока
иначе напечатать сообщение
все если
Закрыть файл my_file

Примечание:  my_file - файловая переменная
             arr - массив целых чисел
             n - размерность массива arr
             nb - размерность массива binarycod
             number - число из массива a
             sum - сумма числа number
             response - признак четности sum
             binarycod - массив двоичного кода числа number
             reverse - число number в обратном порядке следования
             i - подсчет итераций для обращения к элементу массива
```