

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ЭЛЕКТРОТЕХНИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА «ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
АВТОМАТИЗИРОВАННЫХ СИСТЕМ»

ОТЧЁТ ПО ВЫПОЛНЕННОЙ ЗАДАЧЕ

Дисциплина: «Основы алгоритмизации и программирования»

Выполнил:
Студент группы ИВТ-21-26
Безух Владимир Сергеевич

Пермь,
Сентябрь 2021

Постановка задачи

В старояпонском календаре принят шестидесятилетний цикл, который состоит из пяти двенадцатилетних подциклов. Подциклы обозначаются элементами стихий: дерево, огонь, земля, металл, вода. Внутри двенадцатилетнего подцикла годы носят названия животных: крысы, быка, тигра, кролика, дракона, змеи, лошади, овцы, обезьяны, курицы, собаки и свиньи. Нужно составить программу, которая читает год и выводит название этого года по старояпонскому календарю.

Анализ задачи

Цикл старояпонского календаря начинается с года деревянной крысы. В нашей эре год деревянной крысы впервые встречается под номером года 4. Таким образом, для корректировки номера года относительно нуля достаточно вычесть 4 из номера года. Эта корректировка упрощает дальнейшие поиски остатков.

Если найти остаток от деления скорректированного номера года на 12, то мы получим числа в диапазоне от -12 до 11. Число от 0 до 11 можно сопоставить с животным соответствующего года (аналогично для отрицательных чисел). Если номер года меньше 4, то остаток от деления скорректированного номера года на 12 будет меньше нуля.

Если найти остаток от деления скорректированного номера года на 10, то мы получим числа в диапазоне от -10 до 9. Пары чисел от 0-1 до 8-9 можно сопоставить с элементами соответствующего года (аналогично для отрицательных пар чисел). Если номер года меньше 4, то остаток от деления скорректированного номера года на 10 будет меньше нуля.

Для хранения последовательности животных и элементов можно воспользоваться массивом строк. Для сопоставления остатков от деления скорректированного номера года на 10 или 12 с индексами соответствующих животных и элементов достаточно:

- прибавить 10 или 12 к одноимённому остатку от деления, если он меньше 0;
- в конечном итоге поделить нацело на 2 остаток от деления на 10.

Остаток от деления скорректированного номера года на 12	Животное	Остаток от деления скорректированного номера года на 10	Элемент
-12, 0	Крыса	-10, -9, 0, 1	Дерево
-11, 1	Бык	-8, -7, 2, 3	Огонь
-10, 2	Тигр	-6, -5, 4, 5	Земля
-9, 3	Кролик	-4, -3, 6, 7	Металл
-8, 4	Дракон	-2, -1, 8, 9	Вода
-7, 5	Змея		
-6, 6	Лошадь		
-5, 7	Овца		
-4, 8	Обезьяна		
-3, 9	Курица		
-2, 10	Собака		
-1, 11	Свинья		

Описание переменных

const string ANIMALS[12] = {"крысы!", "быка!", "тигра!", "кролика!", "дракона!", "змеи!", "лошади!", "овцы!", "обезьяны!", "курицы!", "собаки!", "свины!"} — массив строк для удобного хранения последовательности животных.

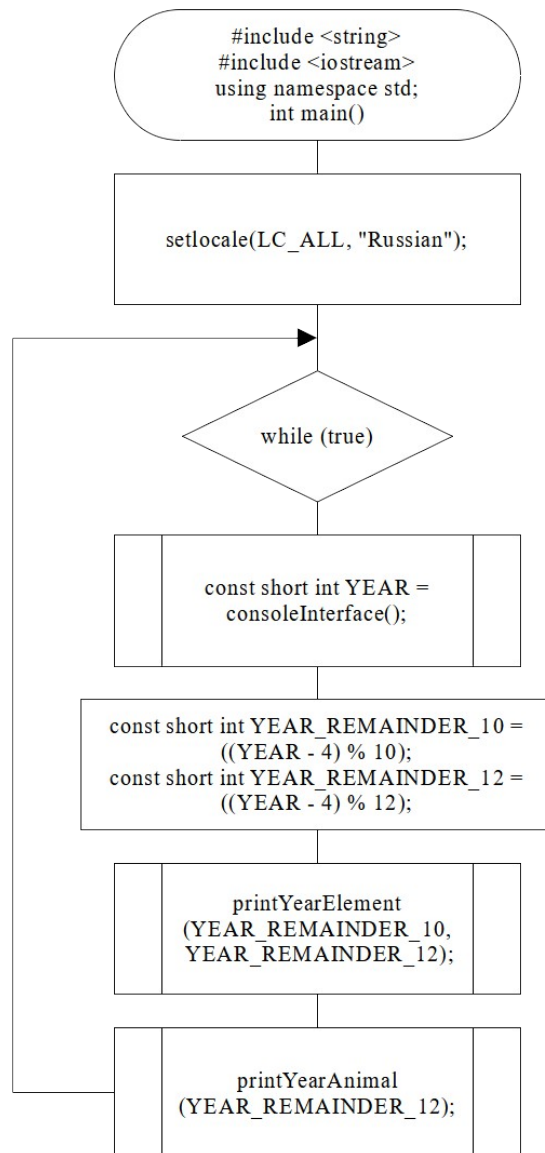
const string ELEMENTS[5] = {"деревянный", "огненный", "земляной", "металлический", "водяной"} — массив строк для удобного хранения последовательности элементов.

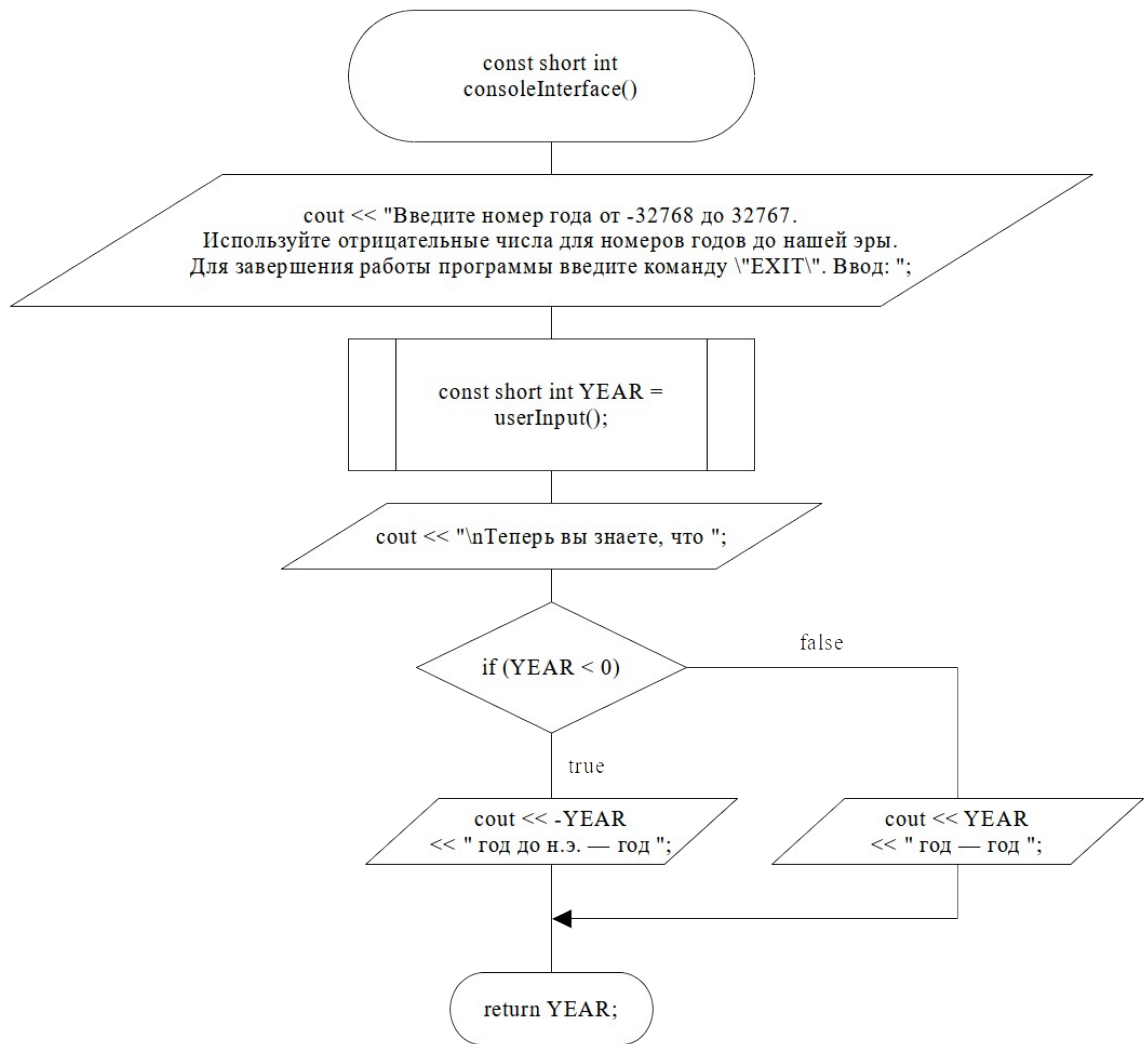
const short int YEAR — 16-битное целое число, номер года от -32768 до 32767. Если после проверки введённых данных выясняется, что пользователь ввёл целое 16-битное число, то это число присваивается к константой целочисленной переменной YEAR.

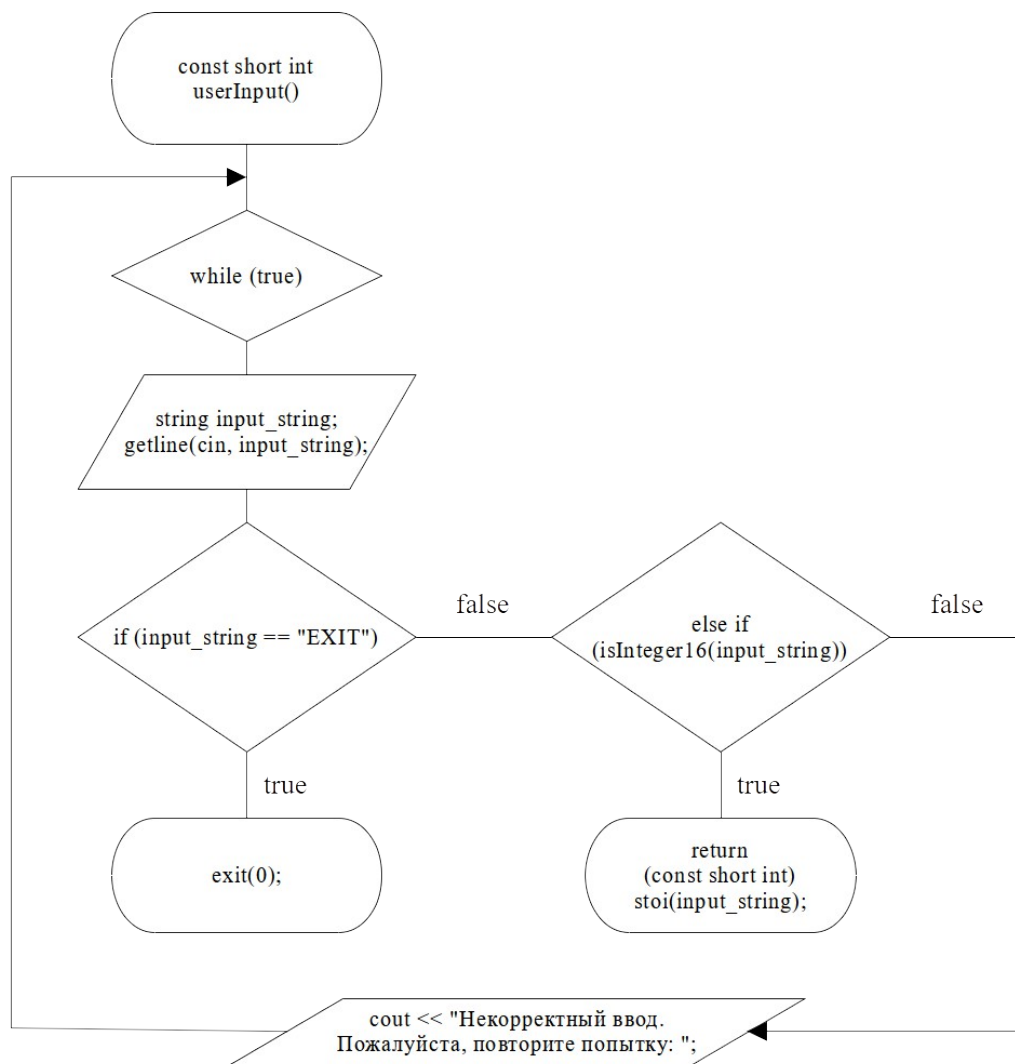
const short int YEAR_REMAINDER_10 — 16-битное целое число, остаток от деления скорректированного номера года на 10. В переменной хранится результат выражения $((YEAR - 4) \% 10)$.

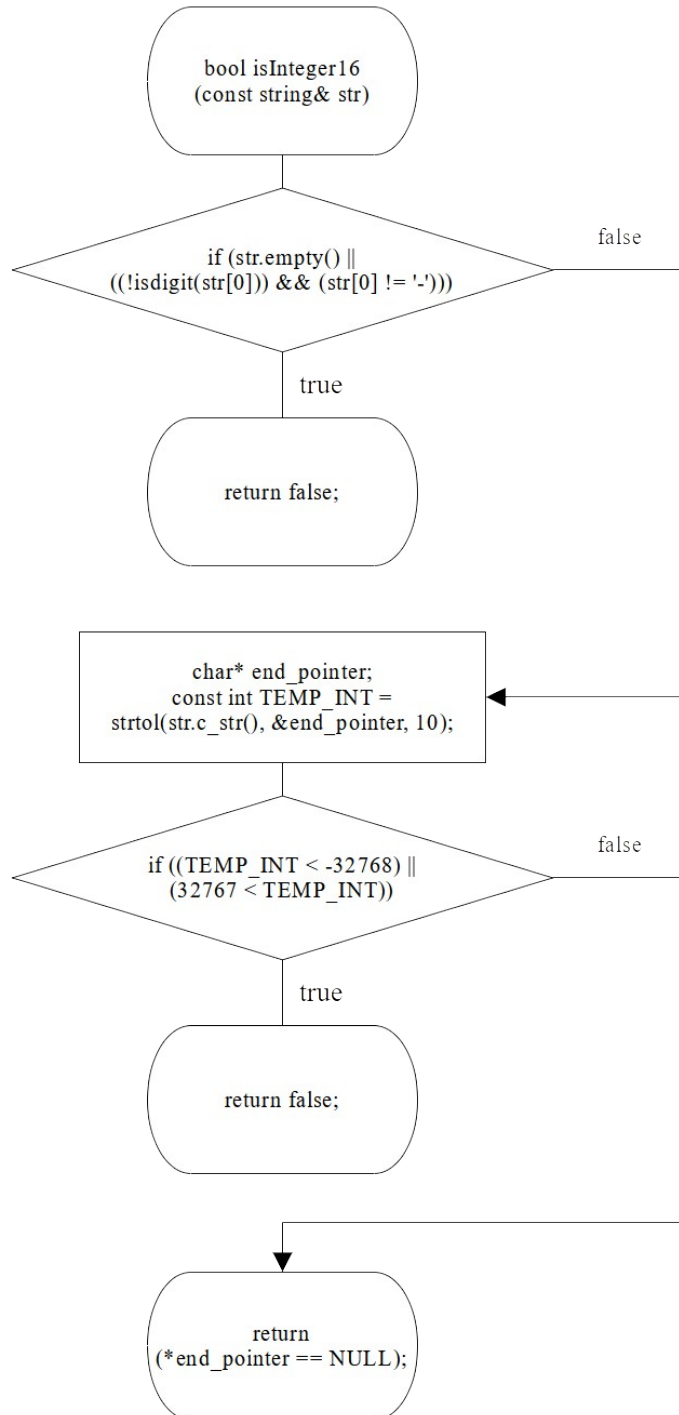
const short int YEAR_REMAINDER_12 — 16-битное целое число, остаток от деления скорректированного номера года на 12. В переменной хранится результат выражения $((YEAR - 4) \% 12)$.

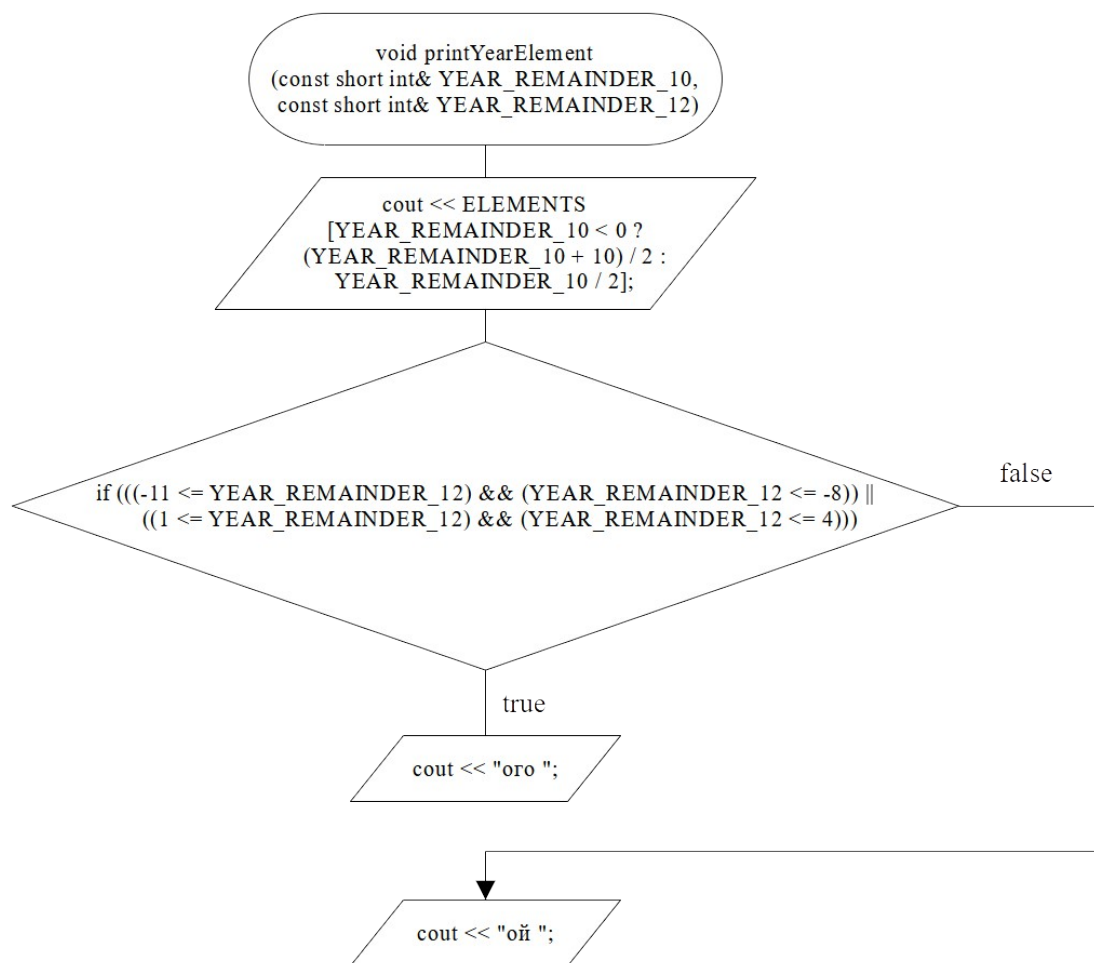
Блок-схема











void printYearAnimal
(const short int& YEAR_REMAINDER_12)

cout << ANIMALS
[YEAR_REMAINDER_12 < 0 ?
YEAR_REMAINDER_12 + 12 :
YEAR_REMAINDER_12]
<< "\n\n";

Исходный код

```
/*
 * This code is licensed under the Creative Commons
 * Attribution - NonCommercial - NoDerivatives 4.0 International License.
 * To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-nd/4.0/
 * or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.
 *
 * https://bezukh.wixsite.com/blog
 * https://github.com/BezukhVladimir
 *
 * © Developed by Bezukh Vladimir. All right reserved.
 */

/*
 * Developed by Bezukh Vladimir
 * September 2021
 * ИБТ-21-26
 *
 * Консольное приложение для определения названий годов по старояпонскому календарю.
 * Основной алгоритм – обыкновенный эвристический. Работает для годов до нашей эры.
 * Реализована проверка вводимых пользователем данных.
 * Программа работает в "while (true)" цикле,
 * что позволяет проверить неограниченное число номеров годов.
 * Для завершения работы программы достаточно ввести команду "EXIT".
 */

#include <string>
#include <iostream>

using namespace std;

const string ANIMALS[12] = {"крысы!", "быка!", "тигра!", "кролика!", "дракона!", "змеи!",
                           "лошади!", "овцы!", "обезьяны!", "курицы!", "собаки!", "свиньи!"};
const string ELEMENTS[5] = {"деревянн", "огненн", "землян", "металлическ", "водян"};

bool isInteger16(const string& str)
{
    /* проверка на пустую строку, проверка первого символа строки */
    if (str.empty() || (!isdigit(str[0]) && (str[0] != '-'))
        return false;

    char* end_pointer;
    const int TEMP_INT = strtol(str.c_str(), &end_pointer, 10);

    if ((TEMP_INT < -32768) || (32767 < TEMP_INT)) /* проверка 16 битного диапазона */
        return false;

    return (*end_pointer == NULL); /* если строка – целое число,
                                   то указатель будет равен нулю-терминатору */
}
```

```

const short int userInput()
{
    while (true)
    {
        // считываем строку, введенную пользователем
        string input_string; getline(cin, input_string);

        if (input_string == "EXIT") // команда для выхода из программы
            exit(0);
        else if (isInteger16(input_string)) // если строка – целое 16 битное число, то...
            return (const short int) stoi(input_string);
        /* ...конвертируем строку в целое число,
           преобразуем в константное 16 битное число */
        else
            cout << "Некорректный ввод. Пожалуйста, повторите попытку: ";
    }
}

const short int consoleInterface()
{
    cout << "Введите номер года от -32768 до 32767. Используйте
            отрицательные числа для номеров годов до нашей эры. Для завершения
            работы программы введите команду \"EXIT\". Ввод: ";

    const short int YEAR = userInput();

    cout << "\nТеперь вы знаете, что ";

    if (YEAR < 0) // динамический пользовательский интерфейс
        cout << -YEAR << " год до н.э. – год ";
    else
        cout << YEAR << " год – год ";

    return YEAR;
}

void printYearElement(const short int& YEAR_REMAINDER_10,
                     const short int& YEAR_REMAINDER_12)
{
    /* найдя остаток от деления скорректированного относительно нуля
       номера года на 10, получим числа от -10 до 9: значения соответственно
       равны определенным элементам годов в старояпонском календаре */

    /* корректировка значения для соответствия с нужным индексом массива ELEMENTS:
       – если значение меньше нуля, то прибавляем к нему 10, а затем делим нацело на 2;
       – иначе сразу делим значение нацело на 2. */

    cout << ELEMENTS[YEAR_REMAINDER_10 < 0 ?
                     (YEAR_REMAINDER_10 + 10) / 2 : YEAR_REMAINDER_10 / 2];

    /* найдя остаток от деления скорректированного относительно нуля
       номера года на 12, получим числа от -12 до 11: так можно узнать
       животное для этого номера года в старояпонском календаре */

    if (((-11 <= YEAR_REMAINDER_12) && (YEAR_REMAINDER_12 <= -8)) ||
        ((1 <= YEAR_REMAINDER_12) && (YEAR_REMAINDER_12 <= 4)))
        cout << "ого ";
    else
        cout << "ой ";
}

```

```

void printYearAnimal(const short int& YEAR_REMAINDER_12)
{
    /* найдя остаток от деления скорректированного относительно нуля
    номера года на 12, получим числа от -12 до 11: так можно узнать
    животное для этого номера года в старояпонском календаре */

    /* корректировка значения для соответствия с нужным индексом массива ANIMALS:
    – если значение меньше нуля, то прибавляем к нему 12;
    – иначе оставляем без изменений. */

    cout << ANIMALS[YEAR_REMAINDER_12 < 0 ?
                    YEAR_REMAINDER_12 + 12 : YEAR_REMAINDER_12] << "\n\n";
}

int main()
{
    setlocale(LC_ALL, "Russian");

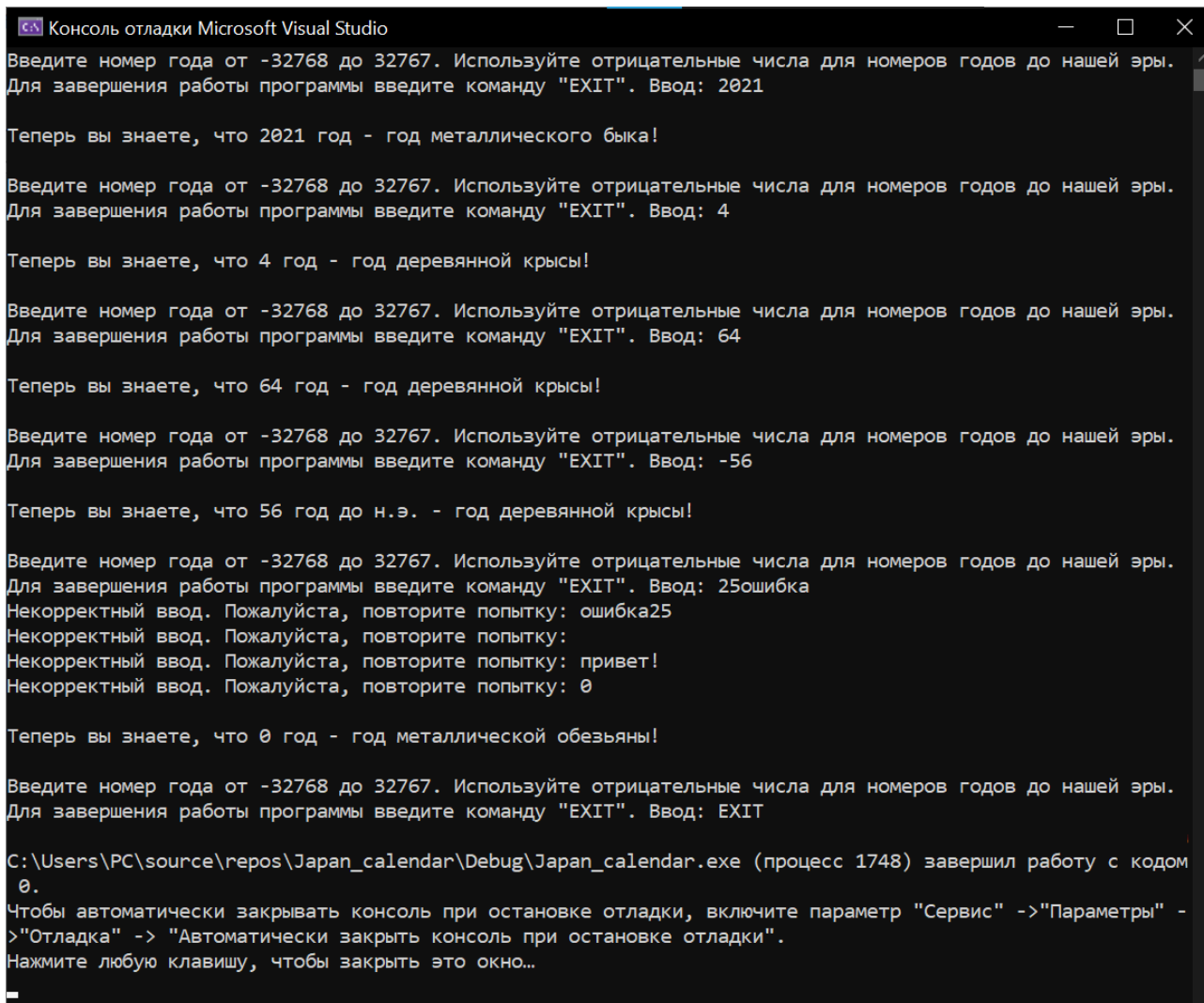
    while (true)
    {
        const short int YEAR = consoleInterface();

        /* остаток от деления на 10 и 12 с учётом корректировки старояпонского
        60-летнего цикла относительно нулевого года нашей эры */
        const short int YEAR_REMAINDER_10 = ((YEAR - 4) % 10);
        const short int YEAR_REMAINDER_12 = ((YEAR - 4) % 12);

        /* поиск названия года старояпонского календаря через эвристические алгоритмы */
        printYearElement(YEAR_REMAINDER_10, YEAR_REMAINDER_12);
        printYearAnimal(YEAR_REMAINDER_12);
    }
}

```

Скриншот консольного интерфейса программы



```
Консоль отладки Microsoft Visual Studio
Введите номер года от -32768 до 32767. Используйте отрицательные числа для номеров годов до нашей эры.
Для завершения работы программы введите команду "EXIT". Ввод: 2021

Теперь вы знаете, что 2021 год - год металлического быка!

Введите номер года от -32768 до 32767. Используйте отрицательные числа для номеров годов до нашей эры.
Для завершения работы программы введите команду "EXIT". Ввод: 4

Теперь вы знаете, что 4 год - год деревянной крысы!

Введите номер года от -32768 до 32767. Используйте отрицательные числа для номеров годов до нашей эры.
Для завершения работы программы введите команду "EXIT". Ввод: 64

Теперь вы знаете, что 64 год - год деревянной крысы!

Введите номер года от -32768 до 32767. Используйте отрицательные числа для номеров годов до нашей эры.
Для завершения работы программы введите команду "EXIT". Ввод: -56

Теперь вы знаете, что 56 год до н.э. - год деревянной крысы!

Введите номер года от -32768 до 32767. Используйте отрицательные числа для номеров годов до нашей эры.
Для завершения работы программы введите команду "EXIT". Ввод: 25ошибка
Некорректный ввод. Пожалуйста, повторите попытку: ошибка25
Некорректный ввод. Пожалуйста, повторите попытку:
Некорректный ввод. Пожалуйста, повторите попытку: привет!
Некорректный ввод. Пожалуйста, повторите попытку: 0

Теперь вы знаете, что 0 год - год металлической обезьяны!

Введите номер года от -32768 до 32767. Используйте отрицательные числа для номеров годов до нашей эры.
Для завершения работы программы введите команду "EXIT". Ввод: EXIT

C:\Users\PC\source\repos\Japan_calendar\Debug\Japan_calendar.exe (процесс 1748) завершил работу с кодом
0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" -
>"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```