**Постановка задачи:**

Вывести на экран показания часов.

Исходные данные – часы и минуты в формате целых чисел. Результат должен быть представлен в смешанной текстово-числовой форме с учетом падежных окончаний.

**Разработка алгоритма:**

Алгоритм переводит 24-часовой формат исходных данных (числовых данных типа int) в 12-часовой формат c сохранением времени суток, а также выводит слова час и минута в нужном формате. формат исходных данных (числовых данных типа int) в 12-часовой формат таким образом. В программе содержится 3 переменных: h -часы (тип int), m-минуты (тип int), mm – показывает, равняется ли количество минут 0 или нет (тип bool) и hh для определения времени суток (int). Также есть переменная input для повторного ввода с консоли.

На вход поступает сначала количество часов (записывается в переменную h), которое проверяется на корректность, затем количество минут (записывается в переменную m), которое также проверяется на корректность, а также проверяется равняется ли значение минут 0, если да, то в переменную mm записывается значение Истина.

Затем, проверяется два значения времени, для которых предусмотрена особая форма: 12 00 – полдень, 00 00 – полночь. Для этого проверяется истинность переменной mm, а затем проверяется равенство переменной h(часы) 12-ти (полдень) и 0 (полночь). Затем поэтапно проводится построение стандартной формы записи времени в 12-часовом формате. В зависимости от введенных данных можно получить следующие результаты.

Пример:

Введите часы и нажмите на клавишу enter

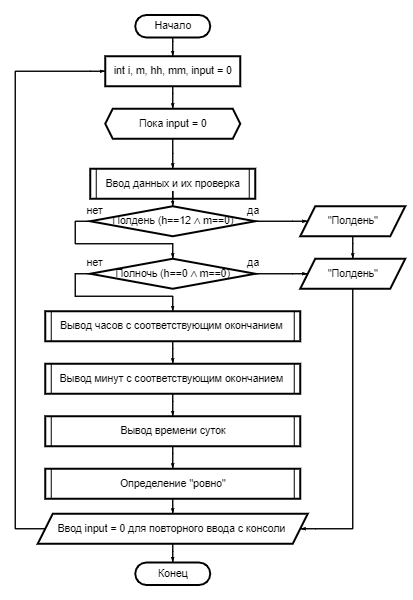
12

Введите минуты и нажмите на клавишу enter

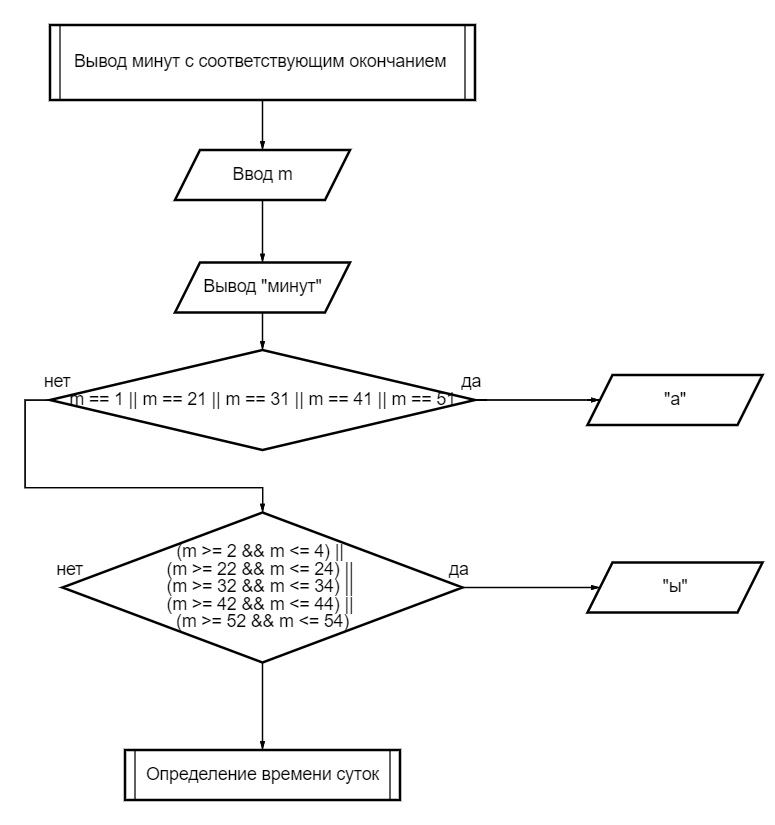
14

12 часов 14 минут дня

Для дальнейшего ввода нажмите 0



**Обработка минут**

****

**Текст программы:**

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_CTYPE, "Russian");

int input = 0;

int h, m, hh;

bool mm;

while (input == 0) {

cout << "Введите часы и нажмите на клавижу enter" << endl;

//Ввод данных

cin >> h;

//Проверка часов

if (h < 0 || h > 23) {

cout << "Неверно введены часы " << endl;

}

else {

cout << "Введите минуты и нажмите на клавишу enter" << endl;

cin >> m;

mm = m == 0;

//Проверка минут

if (m > 59 || m < 0)

{

cout << "Неверно введены минуты " << endl;

}

else

{

//Если полдень

if (h == 12 && mm)

{

cout << "Полдень" << endl;

}

else

{

//Если полночь

if (h == 0 && mm)

{

cout << "Полночь" << endl;

}

else

{

if (h >= 0 && h < 5)

{

hh = 1; //ночь

}

if (h >= 5 && h < 12)

{

hh = 2; // утра

}

if (h >= 12 && h < 18)

{

hh = 3; // дня

}

if (h >= 18 && h < 24)

{

hh = 4; // вечера

}

if (h > 12) {

h = h - 12;

}

//Вывод часов с соответствующим окончанием

cout << h;

cout << " час";

if (h == 1)

{

}

else

{

if ((h >= 2 && h <= 4))

{

cout << "а";

}

if ((h >= 5 && h <= 20) || (h == 0))

{

cout << "ов";

}

}

//Вывод минут с соответствующим окончанием

if (!mm)

{

cout << " " << m << " минут";

if (m == 1 || m == 21 || m == 31 || m == 41 || m == 51)

cout << "а";

else if ((m >= 2 && m <= 4) || (m >= 22 && m <= 24) || (m >= 32 && m <= 34) || (m >= 42 && m <= 44) || (m >= 52 && m <= 54))

cout << "ы";

else

{

}

}

//Определение времени суток

if (hh == 1)

{

cout << " ночи" << endl;

}

if (hh == 2)

{

cout << " утра" << endl;

}

if (hh == 3)

{

cout << " дня" << endl;

}

if (hh == 4)

{

cout << " вечера" << endl;

}

//Определение "ровно"

if (mm)

{

cout << " ровно" << endl;

}

}

}

}

}

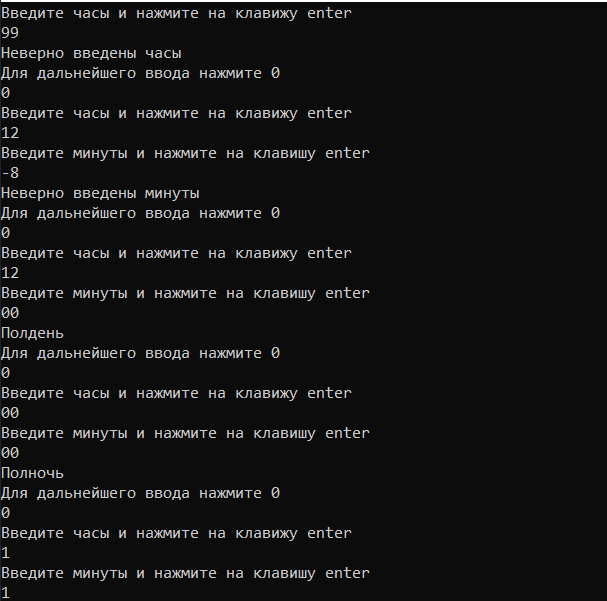
cout << "Для дальнейшего ввода нажмите 0" << endl;

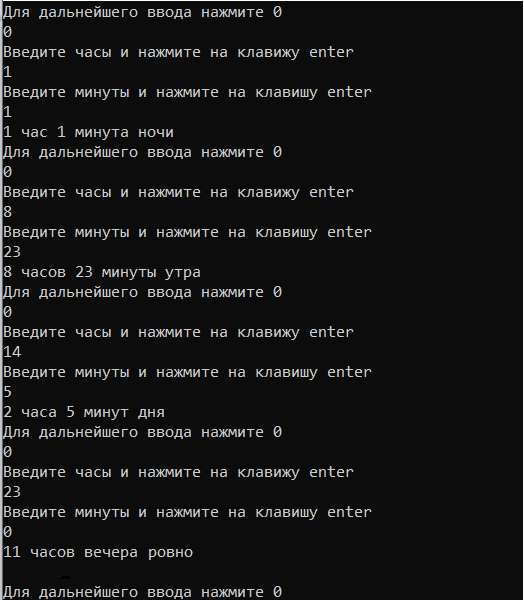
cin >> input;

}

}

**Результаты:**

****

****

**Вывод:**

Я сделала программу, которая выводит показания часов, переводя их из 24-часового формата в 12-часовой с сохранением времени суток. В ходе создания программы я смогла понять преимущества и недостатки условного оператора if.