**Карта урока для организации занятий с использованием**

**электронного обучения и дистанционных образовательных технологий**

|  |  |
| --- | --- |
| **Учитель** | Савушкина Т.Р. |
| **Предмет** | Информатика |
| **Класс** | 8а |
| **Дата проведения урока** | 16.05.2020 |
| **Тема урока** | Конспект урока "Различные варианты программирования циклического алгоритма" |
| **Основные изучаемые вопросы** | ﻿ Вопросы:  ·     Способы организации циклов при решении задач |
| **Ссылка на эл. платформу** | ﻿﻿ <https://join.skype.com/bWlShxFG2fjf> |
| **Тип урока** | Онлайн |
| **Форма обратной связи** | На электронную почту [devon77@yandex.ru](mailto:devon77@yandex.ru) |
| **Задания** | |
| Разбор домашнего задания | Написать программу, которая возводит в 3 степень числа от 1 до 9 , а затем находит их сумму |
| Практическая работа | **Задача:** Определить, является ли целое число **n** простым. Число **n** пользователь вводит с клавиатуры. 4 ≤ **n** ≤ 2 000 000 000.  Простым называется целое число, которое не делится нацело ни на одно целое число, кроме единицы и себя самого. Для того, чтобы определить, является ли число **n** простым, нам достаточно проверить, не делится ли оно на какое-нибудь из целых чисел на промежутке от 2 до .  Сперва рассмотрим вариант решения этой задачи при помощи цикла с параметром. Составим блок-схему алгоритма решения задачи. Сперва считаем значение **n** введённое пользователем. Для того, чтобы определить, является ли **n**простым, нам понадобится логическая переменная **p**. Присвоим ей значение «Истина», предположив, что **n** – простое число. Дальше будет следовать цикл для **i** от 2 до , а в нём – условный оператор, который будет проверять, равен ли остаток от деления **n** на **i** 0. Если это условие будет выполняться, значит число **n** не является простым – и мы присвоим переменной **p**, значение «Ложь», в противном случае ничего делать не нужно. После цикла будет следовать условный оператор, который будет проверять значение переменной **p**. Если значение **p** – «Истина», значит **n** – простое число, выведем на экран поясняющее сообщение об этом. Если же значение **p** – «ложь», выведем на экран поясняющее сообщение о том, что число **n**не является простым.  Блок-схема алгоритма  Теперь напишем программу по составленной блок-схеме. Назовём её prostoe\_chislo. Для решения задачи нам понадобятся две целочисленные переменные: **n** и **i**. Так как **n** находится на промежутке от 4 до 2 000 000 000, а **i** изменяется в пределах от 2 до , объявим их принадлежащими целочисленному типу **integer**. Так же нам понадобится логическая переменная **p** типа **boolean**.  Запишем логические скобки. Программа будет начинаться с оператора **writeln**, который будет выводить на экран поясняющее сообщение о том, что это программа, определяющая, является ли число **n** простым. Дальше будет следовать ещё один оператор **write**, который будет выводить на экран запрос на ввод числа **n**. После него будет оператор **readln**, считывающий значение переменной **n**. Затем будет оператор присваивания переменной **p** значения **true**. Теперь запишем цикл для **i** от 2 до .  Квадратный корень из **n** извлекается с помощью функции **sqrt**. Так как эта функция возвращает значение вещественного типа, поэтому при вычислении результата могут быть погрешности, поэтому прибавим к корню из **n** единицу. Так же в цикле должно быть указано значение целого типа, приведём значение вещественного типа к целому с помощью функции округления **round**. Тело цикла будет содержать всего один условный оператор, поэтому логические скобки не требуются. Запишем условный оператор, его условием будет то, что остаток от деления **n** на **i** равен нулю. После служебного слова **then**будет следовать оператор присваивания переменной **p** значения false.  После цикла запишем условный оператор. В качестве его условия зададим значение логической переменной **p**. Она будет содержать значение истинности утверждения о том, что **n** – простое число. После слова **then** в этом условном операторе будет следовать оператор **write**, который выводит поясняющее сообщение о том, что **n** – простое число. После слова **else**будет следовать оператор **write**, выводящий на экран сообщение о том, что **n** не является простым числом.  **program**prostoe\_chislo;  **var**  n, i: integer;   p: boolean;  **begin**  writeln ('Программа, определяющая, является ли целое число n простым.');   write ('n=');   readln (n);   p:=true;  **for**i:=2 **to**round (sqrt(n)+1) **do**  **if**n **mod**i=0  **then**p:=false;  **if**p  **then**write ('Число ', n, ' является простым.')  **else**write ('Число', n, ' не является простым.');  **end**.  Исходный код программы  Запустим программу на выполнение и зададим **n** = 33.  Число 33 действительно не является простым, так как делится нацело на 3. Снова запустим программу на выполнение и введём число **n** = 79.  Это число действительно является простым так как делится нацело лишь на себя и на 1. |
|  |  |
| **Домашнее задание** | **Просмотреть по возможности дополнительный материал**  <https://youtu.be/K8Jxw0-loBo> |
|  |  |

Вопросы можно задать по адресу электронной почты

или в мессенджерах: WhatsApp (№\_89043417942) или в онлайн формате по ссылке

с 13.00 до 14.00 *(часы неаудиторной занятости,* *проведение индивидуальной*

*консультации)*

Выполненное практическое задание необходимо предоставить в любом доступном формате (скан, фотография,; *указывается вариант, которым владеет учитель и учащиеся (группы учащихся))*:

письмом на адрес электронной почты для обратной связи [devon77@yandex.ru](mailto:devon77@yandex.ru)

* сообщением в WhatsApp №\_89043417942

При отправке ответа в поле «Тема письма» и названии файла укажите свои данные: **класс,** **учебный предмет, фамилию, имя и отчество.**