

К. Н. Костицин

Хорошевская школа, г. Москва, Россия

П. К. Чернова

Школа № 1293, г. Москва, Россия

О ПОДХОДАХ К ОБУЧЕНИЮ ПРОГРАММИРОВАНИЮ В РОССИИ И ЗА РУБЕЖОМ

Аннотация

В статье описан опыт обучения программированию в учебных программах школ Ирландии и провинции Онтарио (Канада). Провинция Онтарио — успешная участница Международного исследования компьютерной и информационной грамотности (*англ.* International Computer and Informational Literacy Study — ICILS); Ирландия придает особое значение возможности формирования в курсе информатики навыков XXI века. Представлены цели, задачи, структура содержания соответствующих учебных курсов и планируемые результаты обучения. Сопоставлены канадский, ирландский и российский подходы к обучению программированию в школе по следующим критериям: обязательность освоения, тематические блоки, программное обеспечение, предметные результаты, навыки XXI века, продолжительность курса. Описаны достоинства российского курса информатики и возможности использования зарубежного опыта при построении курсов по выбору и/или программ внеурочной деятельности.

Ключевые слова: программирование, алгоритмизация, учебная программа, основная школа, информатика, Junior Cycle, Coding.

DOI: 10.32517/2221-1993-2021-20-7-20-26

1. Введение

В наши дни в связи с повсеместным использованием информационных технологий навыки программирования становятся общезначимыми в различных профессиональных сферах. С программированием на протяжении последних нескольких десятилетий так или иначе знакомят школьников во многих уголках мира [13], преследуя при этом следующие основные цели: формировать у обучающихся навыки решения проблем и логического мышления [2, 5, 7], которые являются ключевыми навыками XXI века; реагировать на нехватку квалифицированной рабочей силы в области ИТ [2].

В данной статье будут рассмотрены подходы к обучению программированию в Канаде (провинция Онтарио) и Ирландии, выделены параметры сравнения и на основе них дана сопоставительная характеристика

зарубежных и отечественного подходов. Выбор стран не случаен: канадская провинция Онтарио — успешная участница Международного исследования компьютерной и информационной грамотности (*англ.* International Computer and Informational Literacy Study — ICILS), использующая для достижения высоких результатов все доступные информационные технологии; Ирландия придает особое значение возможности формирования в курсе информатики навыков XXI века.

2. Обучение программированию в школах провинции Онтарио, Канада

Онтарио является самой большой по численности населения провинцией Канады. Школьная система Онтарио состоит из двух ступеней: начальная (I—VIII классы: 6—13 лет) и средняя (IX—XII классы: 14—17 лет) [12].

Контактная информация

Костицин Кирилл Николаевич, учитель информатики, Хорошевская школа, г. Москва, Россия; *адрес:* 123423, Россия, г. Москва, ул. Народного ополчения, д. 9А; *e-mail:* kostitsin.kirill@yandex.ru

Чернова Полина Константиновна, учитель математики и информатики, школа № 1293, г. Москва, Россия; *адрес:* 121552, Россия, г. Москва, Оршанская ул., д. 10, корп. 1; *e-mail:* pk.chernova@gmail.com

K. N. Kostitsin

HoroSchool, Moscow, Russia

P. K. Chernova

School 1293, Moscow, Russia

NATIVE AND FOREIGN APPROACHES TO TEACHING CODING AT SCHOOL

Abstract

The article describes the experience of teaching coding in the curricula of schools in Ireland and the province of Ontario (Canada). The province of Ontario is a successful participant in the International Computer and Informational Literacy Study (ICILS); Ireland attaches particular importance to the possibility of developing 21st century skills in coding. The goals, objectives, structure of the content of the corresponding training courses and the planned learning outcomes are presented. The Canadian, Irish and Russian approaches to teaching coding at school are compared according to the following criteria: compulsory mastery, thematic blocks, software, subject results, skills of the XXI century, course duration. The advantages of the Russian informatics course and the possibility of using foreign experience in the construction of elective courses and/or extracurricular programs are described.

Keywords: coding, programming, algorithmization, school curriculum, basic school, Computer Science, informatics, Junior Cycle.

Одной из содержательных линий предмета «Наука и технологии» (I—VIII классы) является линия «Структуры и механизмы», в которой информатика представлена как средство изучения окружающей действительности. В рамках этой линии ученики начальных классов получают исключительно пользовательские навыки работы на компьютере, знакомятся с 3D-моделированием (TinkerCad) для создания механизмов и робототехникой (LEGO MINDSTORM) для получения базовых представлений об алгоритмизации [8]. В IX классе школьники сдают экзамены и выбирают степень углубленности общеобразовательных предметов на X—XII классы.

В X—XII классах в школах Онтарио изучается предмет «Компьютерные науки» (Computer Science).

Его основная цель — сформировать у обучающихся знания, умения и навыки, востребованные как в школе, так и за ее пределами.

В курсе освещаются **четыре содержательные линии**:

- разработка программного обеспечения (включая управление проектами);
- алгоритмы и структуры данных;
- корректность и эффективность программы;

• профессиональная и этическая ответственность. Курс «Компьютерные науки» в школах провинции Онтарио является непрерывным [8].

Учащемуся предлагается выбрать **тип курса**, исходя из его интересов, достижений и планов на будущее:

- подготовка к поступлению в университет (обучение реализуется под руководством наставников из университетов);
- подготовка к поступлению в колледж (программа курса составлена с учетом требований к поступлению на большинство программ колледжей, на стажировку компаний или другие программы обучения, с гарантированным трудоустройством у работодателя);
- открытый курс (программа курса направлена на расширение знаний и навыков учащихся по предметам, которые отражают их интересы, на подготовку школьников к активному и полезному участию в жизни общества).

В таблице 1 представлены цели и задачи курса «Компьютерные науки», ориентированного на подготовку школьников к продолжению обучения в университете («университетского» курса).

Таблица 1

Цели и задачи курса «Компьютерные науки»

Класс	Тема	Цели	Задачи (учащийся должен уметь)
X	Введение в программирование	1. Описывать фундаментальные концепции и конструкции программирования	1.1. Использовать правильную терминологию для описания концепций программирования. 1.2. Описывать типы данных, которые компьютеры могут обрабатывать и хранить. 1.3. Объяснять разницу между константами и переменными, используемыми в программировании. 1.4. Определять выражения и последовательность инструкций в программировании, принимая во внимание порядок операций. 1.5. Определять ситуации, в которых необходимы циклические конструкции. 1.6. Описывать функцию булевых операторов, операторов сравнения.
		2. Планировать и разрабатывать простые программы, используя фундаментальные концепции программирования	2.1. Использовать визуальную модель решения проблем для планирования содержания программы. 2.2. Использовать переменные, выражения и операторы присваивания для хранения и управления числами и текстами в программе (например, в программе викторины, в программе преобразования единиц измерения). 2.3. Разрабатывать программы, которые включают классические алгоритмы (например, игры «Угадай число», «Камень — ножницы — бумага», множественный выбор, викторина). 2.4. Разрабатывать программы, эффективно использующие циклические структуры (например, простая анимация, подбрасывание монеты). 2.5. Объяснять разницу между синтаксическими ошибками, логическими ошибками и ошибками времени выполнения программы. 2.6. Сравнить разные программные среды для решения одних и тех же проблем и сопоставлять их со сферой применения (например, решение, разработанное на языке программирования, по сравнению с решением, разработанным с использованием электронной таблицы).
XI	Концепции и навыки программирования	1. Демонстрировать возможность использования в компьютерных программах различных типов данных, включая одномерные массивы	1.1. Правильно использовать в компьютерных программах константы и переменные, включая целые числа, числа с плавающей запятой, строки и логические значения. 1.2. Демонстрировать понимание того, как компьютер использует различные системы (ASCII, Unicode) для внутреннего представления данных и хранения информации. 1.3. Описывать структуру одномерного массива и связанных с ним данных, включая элементы, индексы и границы. 1.4. Разрабатывать программы, которые объявляют, инициализируют, изменяют одномерные массивы данных и получают доступ к ним.

Продолжение табл. 1

Класс	Тема	Цели	Задачи (учащийся должен уметь)
		2. Демонстрировать умение использовать подпрограммы (процедуры и функции) в компьютерных программах	2.1. Демонстрировать способность использовать существующие подпрограммы (например, генератор случайных чисел, подстроку, абсолютное значение) в компьютерных программах. 2.2. Разрабатывать подпрограммы (например, функции, процедуры), которые используют передачу параметров и соответствующую область действия переменных (например, локальную, глобальную) для выполнения задач в программах.
		3. При создании компьютерных программ использовать надлежащие методы обслуживания кода и пользовательские соглашения	3.1. Демонстрировать способность выявлять и исправлять в компьютерных программах синтаксические ошибки, логические ошибки и ошибки времени выполнения. 3.2. Правильно использовать рабочие и профессиональные соглашения (например, именование, отступы, комментарии) для написания программ и внутренней документации. 3.3. Демонстрировать способность интерпретировать ошибки по сообщениям, отображаемым средствами программирования (например, компилятором, средством отладки) в разное время в процессе разработки программного обеспечения (например, при написании, компиляции, тестировании). 3.4. Использовать метод трассировки для понимания потока программы, а также для определения и исправления логических ошибок и ошибок времени выполнения. 3.5. Демонстрировать способность валидировать программу, используя полный набор тестовых примеров.
	Разработка программного обеспечения	1. Использовать различные стратегии решения проблем для различных типов задач самостоятельно и в команде	1.1. Использовать различные стратегии решения проблем (например, пошаговое уточнение, «разделяй и властвуй», обратная связь, примеры, крайние случаи, таблицы и диаграммы, метод проб и ошибок) при решении различных типов задач. 1.2. Демонстрировать способность решать проблемы самостоятельно и в составе команды.
		2. Использовать алгоритмы проектирования согласно техническому заданию	2.1. Разрабатывать простые алгоритмы (например, добавлять данные в отсортированный массив, удалять элементы из середины массива) согласно спецификации. 2.2. Решать общие проблемы (например, расчет гипотенузы, определение простых чисел, расчет площади и длины окружности) путем применения математических уравнений или формул в алгоритме. 2.3. Разрабатывать алгоритмы, обнаруживающие и обрабатывающие исключения (например, деление на ноль).
		3. Применять модель жизненного цикла разработки программного обеспечения к проекту разработки программного обеспечения	3.1. Описывать фазы (определение проблемы, анализ, проектирование, написание кода, тестирование, реализация, сопровождение), этапы (например, дата завершения спецификации программы) и продукты (например, спецификация, блок-схема, программа, документация, отчеты об ошибках) жизненного цикла разработки программного обеспечения. 3.2. Использовать различные методы (например, диалог, анкеты, опросы, исследования) для уточнения спецификации программы. 3.3. Использовать план тестирования для тестирования программ (т. е. идентифицировать сценарии тестирования, определять подходящие входные данные, вычислять ожидаемые результаты, записывать фактические результаты и делать вывод — «прошел» или «не прошел») путем сравнения ожидаемых результатов с фактическими. 3.4. Передавать информацию о статусе проекта (например, основные этапы, выполненная работа, невыполненная работа) эффективно в письменной форме на протяжении всего проекта.
	XII	1. Описывать и использовать концепции и принципы модульного программирования при создании компьютерных программ	1.1. Создавать модульную программу, состоящую из нескольких файлов (например, определяемых пользователем классов, библиотек, модулей). 1.2. Использовать концепции модульного дизайна, которые поддерживают повторно используемый код (например, инкапсуляция, наследование, перегрузка метода, переопределение метода, полиморфизм). 1.3. Демонстрировать способность изменять существующий модульный программный код для улучшения функциональности программы.
		2. Проектировать и записывать алгоритмы и подпрограммы для решения множества задач	2.1. Демонстрировать возможность чтения и записи внешнего файла (например, текстового файла, двоичного файла, базы данных) из компьютерной программы. 2.2. Создавать алгоритмы линейного и двоичного поиска для поиска данных в массиве. 2.3. Создавать подпрограммы для вставки и удаления элементов массива.

Окончание табл. 1

Класс	Тема	Цели	Задачи (учащийся должен уметь)
			2.4. Создавать алгоритм сортировки (например, сортировка пузырьком, вставка, выбор) для сортировки данных в массиве. 2.5. Создавать алгоритмы для обработки элементов в двумерных массивах (например, умножение каждого элемента на константу, обмен элементами, умножение матриц, обработка пикселей в изображении). 2.6. Разрабатывать простой и эффективный рекурсивный алгоритм (например, вычислить факториал, перевести числа в слова, выполнить сортировку слиянием, сгенерировать фракталы, выполнить синтаксический анализ).
		3. Использовать правильные методы обслуживания кода при создании компьютерных программ	3.1. Разрабатывать и внедрять формальный план тестирования (например, модульное тестирование, интеграционное тестирование, регрессионное тестирование) для проекта программного обеспечения, чтобы гарантировать правильность программы. 3.2. Создавать полностью документированный программный код в соответствии с отраслевыми стандартами (например, добавлять комментарии к документам, строки к документации, комментарии к блокам, комментарии к строкам). 3.3. Создавать понятную и поддерживаемую внешнюю пользовательскую документацию (например, файлы справки, учебные материалы, руководства пользователя).

3. Обучение программированию в Ирландии

В Ирландии программирование впервые вводится в курсе информатики на ступени младшей школы. **Курс информатики (Coding)** [11] относится к кратким курсам, так как рассчитан на 100 часов, распределенных между тремя годами обучения. Школа может в индивидуальном порядке внести изменения в программу, но, по результатам изучения 50 сайтов младших школ Ирландии, нигде не было обнаружено измененной программы по программированию.

В вводной части программы по информатике [11] поднимается вопрос о том, что в современной жизни компьютерные технологии присутствуют во всех ее сферах. Курс способствует развитию вычислительного и алгоритмического мышления. **Цель курса:** развить у обучающихся умение формулировать проблему, составлять программу решения проблемы, используя языки программирования, создавая игры, приложения, сайты.

В программе сопоставляются планируемые результаты образовательной программы Ирландии и действия в рамках учебного курса по информатике, которые будут способствовать достижению этих результатов

Выделяются следующие **планируемые результаты** [11]. Учащийся:

- анализирует проблему и составляет алгоритм ее решения, подключая математический аппарат;
- описывает, демонстрирует, интерпретирует, прогнозирует и распознает модели и связи;
- подбирает оптимальные современные технологии для решения возникающих задач;
- осуществляет решение задачи от идеи до ее реализации.

Предполагается, что программа младшей школы направлена на развитие восьми ключевых навыков:

- коммуникативных навыков;
- грамотности;
- навыков самоуправления;
- умения оставаться здоровым;
- навыков целеполагания и умозаключения;
- знания точных наук, в частности математики;
- творческой активности;
- навыков сотрудничества и кооперации.

В программе курса информатики представлены компоненты каждого навыка, которые развиваются в рамках курса. В таблице 3 представлены планируемые результаты обучения по курсу [11], сгруппированные по темам внутри трех линий.

Таблица 2

Структура системы образования Ирландии [9]

Ступени образования	Возрастные рамки	Формат итоговой аттестации
Дошкольное образование — Pre-school (не является обязательным)	До 4–5 лет	Не предусмотрена
Начальная школа — Primary School	От 4–5 до 12–13 лет	Не предусмотрена
Младшая школа — Junior Cycle (аналогично основной школе в РФ)	От 12–13 до 15–16 лет	Экзамены по 10–11 предметам для получения аттестата Junior Certificate
Старшая школа — Senior Cycle (между младшей и старшей школой может быть введен промежуточный год)	От 16–17 до 17–18 лет	Государственный экзамен для получения аттестата Leaving Certificate

Таблица 3

Планируемые результаты курса

Линия	Тема	Учащийся должен уметь/иметь:
<i>Линия 1.</i> Введение в информатику (компьютер как средство решения задач в различных областях; аппаратное обеспечение; основное программное обеспечение; написание кода, тестирование)	<i>Тема 1.</i> Мой цифровой мир: значимость компьютера в современном обществе и для личного использования	1. Объяснять на примерах, в каких сферах используется компьютер и какова его роль в жизни современного человека. 2. Описывать основные компоненты компьютерной системы: процессор, оперативная память, внутреннее хранилище, устройства ввода/вывода, шина. 3. Объяснять процесс выполнения программ, написанных с использованием языка программирования.
	<i>Тема 2.</i> Шаг за шагом на пути к профессии программиста: как начать программировать и разрабатывать базовые алгоритмы	1. Разрабатывать соответствующие алгоритмы, используя псевдокод и/или блок-схемы. 2. Записывать код, реализующий конкретный алгоритм. 3. Описывать основные компоненты и функции структурного программирования (переменные, операторы, циклы, присвоение, модульность). 4. Тестировать код. 5. Оценивать результаты работы в составе группы из двух-трех человек.
<i>Линия 2.</i> Компьютер как средство хранения, обработки и передачи информации	<i>Тема 3.</i> Установление связей: компьютер как средство коммуникации	1. Общее представление об интернете. 2. Описывать процесс передачи информации в интернете и связь компьютеров посредством протоколов. 3. Объяснять, как поисковые системы выполняют запрос. 4. Создавать сайты с использованием HTML и CSS.
	<i>Тема 4.</i> Биты и байты: хранение информации	1. Кодировать информацию с помощью 0 и 1. 2. Описывать способ представления графической информации в компьютерах.
	<i>Тема 5.</i> Проблемы реального мира: компьютерные технологии, которые вдохновляют	1. Определять область информатики, наиболее ему интересную. 2. Проводить исследование по определенной теме/проблеме. 3. Выдвигать тезисы для обсуждения и давать обратную связь. 4. Аргументированно выступать с какой-либо идеей.
<i>Линия 3.</i> Программирование более высокого уровня (более высокий уровень задач по программированию)	<i>Тема 6.</i> Программист: продвинутый уровень программирования и вычислительное мышление	1. Составлять и записывать код для выполнения небольших программных задач с использованием операторов присваивания, арифметических операций, операций сравнения и логических операций. 2. Выполнять небольшие задачи программирования с использованием базовых линейных структур данных (например, массива или списка). 3. Демонстрировать принцип работы функций и/или процедур (их описание и вызов). 4. Записывать программу, в которой используется параллельная или последовательная структура данных, учитывая выбранный язык программирования.
	<i>Тема 7.</i> Документирование кода: документирование и анализ кода	1. Записывать программы (описывать формально) и объяснять их. 2. Представлять программный код в небольших группах. 3. Анализировать код, определять его назначение, описывать возможные ошибки.

Линия 1 «Введение в информатику» обязательно изучается в рамках первого года обучения; две другие линии могут быть изучены в том порядке, который выберет школа.

Программой рекомендовано использовать программное обеспечение, которое находится в открытом доступе и в котором есть возможность просматривать код используемых инструментов.

4. Сопоставление российского и зарубежных курсов информатики

Сопоставим российский курс информатики для VII—IX классов [6], курс «Компьютерные науки» провинции Онтарио [8] и ирландский курс информатики [10] (табл. 4).

Таблица 4

Сравнение подходов к изучению информатики в России, в Ирландии и в Онтарио

№ п/п	Параметр сравнения	Курс информатики для основной школы, Россия	Курс «Компьютерные науки», провинция Онтарио, Канада	Курс информатики, Ирландия
1	Обязательность освоения	Обязательное изучение всеми школьниками	Обязательное изучение всеми школьниками	Изучается по выбору школьников
2	Содержание (основные тематические блоки)	Введение (информация и информационные процессы, компьютер). Математические основы информатики. Алгоритмы и элементы программирования. Использование программных систем и сервисов	Введение в программирование. Концепции и навыки программирования. Разработка программного обеспечения («университетский» курс)	Введение в информатику. Компьютер как средство хранения, обработки и передачи информации. Программирование более высокого уровня
3	Программное обеспечение	Не регламентируется	Не регламентируется	Свободное ПО
4	Предметные результаты	Шире по сравнению с ирландским курсом информатики	Полное совпадение целей на уровне основной школы отечественного курса информатики	В полном объеме перекрываются планируемыми предметными результатами российского курса информатики для основной школы
5	Навыки XXI века	Представлены в формате личностных и метапредметных образовательных результатов	Формируются навыки 4К	Целостная система, состоящая из восьми групп навыков
6	Продолжительность изучения	102 ч	Нет данных	100 ч

5. Выводы

Анализ учебных планов школ канадской провинции Онтарио показывает, что программирование школьники начинают изучать довольно поздно. Отметим, что на уровне целей и задач, представленных в разделе «Введение в программирование» для X класса «университетского» курса «Компьютерные науки» (Computer Science), мы видим полное совпадение с целями (планируемыми результатами), реализуемыми у нас в рамках обучения информатике в основной школе [1], т. е. в более раннем возрасте. Частично на уровне основного общего образования у нас представлен и раздел «Концепции и навыки программирования», изучаемый канадскими школьниками в XI классе [4]. В документах не регламентируются языки программирования для изучения в школе, однако указаны примеры учебных проектов. Содержание «университетского» курса «Компьютерные науки» довольно объемно, но выделяемые часы на изучение тем не указаны.

Говоря об ирландском курсе информатики, приоритетное внимание которого сфокусировано на программировании и направленности на формирование навыков XXI века, можно отметить, что российский курс информатики более широкий и теоретически насыщенный, программирование — его важная, но не единственная часть.

Результаты проведенного нами исследования подтверждают оценку, данную отечественному курсу ин-

форматики для основной школы, в работе Л. Л. Босовой [1]: «Наряду с безусловными достижениями в нашем школьном курсе информатики есть и многочисленные проблемы, связанные в том числе и с минимальным учебным временем, отведенным в учебных планах на информатику, что делает невозможным системное изучение этого предмета, обеспечивающее достижение в полном объеме планируемых результатов, зафиксированных в нормативных документах».

Что касается изучения информатики на уровне среднего общего образования [3], то в российских школах и на базовом, и на углубленном уровнях предусмотрено дальнейшее развитие навыков, скорее, программирования, составления алгоритмов и решения задач, нежели практической разработки программного обеспечения. В противовес этому сильной стороной «университетского» курса в Канаде является профориентационная тема «Разработка программного обеспечения», которая формирует у обучающихся навыки решения проблем, взаимодействия в команде, анализа жизнеспособности проекта. И всё это при поддержке преподавателей вузов. Подходы к обучению школьников разработке программного обеспечения могут быть использованы нашими педагогами при проектировании курсов по выбору и/или программ внеурочной деятельности. Подход ирландских коллег к формированию ряда компетенций и навыков XXI века через информатику и программирование требует дальнейшего изучения и может быть полезен для обогащения методики обучения информатике.

Список использованных источников

1. Босова Л. Л. Достоинства и недостатки отечественного школьного курса информатики // Актуальные проблемы методики обучения информатике в современной школе: Материалы Международной научно-практической интернет-конференции (г. Москва, 24–26 апреля 2018 года). М.: МПГУ, 2018. С. 8–14.
2. Босова Л. Л. О подходах к формированию цифровых навыков обучающихся на уровне общего образования // Международный конгресс по информатике: информационные системы и технологии (в образовании): Материалы международного научного конгресса (г. Минск, 22–23 октября 2020 года). Минск: БГУ, 2020. С. 47–58.
3. Босова Л. Л. Программирование в школе: возможности, проблемы, решения // Информатизация образования — 2018. Труды Международной научно-практической конференции. М.: Академия информатизации образования; Академия компьютерных наук, Институт управления образованием РАО, 2018. С. 172–179.
4. Босова Л. Л., Босова А. Ю. Информатика. Программа для основной школы: 5–6 классы. 7–9 классы. 2-е изд. М.: БИНОМ. Лаборатория знаний, 2014. 88 с.
5. Босова Л. Л., Самылкина Н. Н. Информатика на уровне среднего общего образования: основные подходы к реализации // Профильная школа. 2020. Т. 8. № 4. С. 32–45. DOI: 10.12737/1998-0744-2020-32-45
6. Примерная основная общеобразовательная программа основного общего образования // Реестр примерных основных общеобразовательных программ. <https://fgosreestr.ru/registry/primernaya-osnovnayaobrazovatel'naya-programma-osnovnogo-obshhego-obrazovaniya-3/>
7. Сухас Д., Парандекар Е., Патаракин Г. Обучение детей программированию: залог развития человеческого капитала в XXI веке. Руководство для российских законодателей и практиков в области образования. Яйла — М.: Алекс (ИП Поликанин А. А.), 2019. 164 с.
8. Curriculum Ontario. <http://www.edu.gov.on.ca/eng/curriculum/secondary/subjects.html>
9. Education in the Republic of Ireland // Wikipedia. https://en.wikipedia.org/wiki/Education_in_the_Republic_of_Ireland#Years
10. Framework for Junior Cycle. <https://ncca.ie/en/junior-cycle/framework-for-junior-cycle>
11. Junior Cycle. Short courses: Coding // NCCA. <https://curriculumonline.ie/Junior-cycle/Short-Courses/Coding/>
12. Population and Dwelling Count Highlight Tables, 2011 Census // Canada Statistics. <https://www12.statcan.gc.ca/census-recensement/2011/dp-pd/hltfstfst/pl/Table-Tableau.cfm?LANG=Eng&T=101&S=50&O=A>
13. World Economic Forum Conclusion 2017 // World Economic Forum. https://www3.weforum.org/docs/WEF_EGW_FOJ_MENA.pdf

НОВОСТИ

Владимир Путин поручил наделить Минпросвещения России правом регулирования цифровых образовательных ресурсов

Президент России Владимир Путин поручил Правительству Российской Федерации совместно с комиссией Государственного Совета Российской Федерации по направлению «Образование» обеспечить наделение Министерства просвещения Российской Федерации правом устанавливать порядок экспертизы содержания и порядок лицензирования электронных образовательных ресурсов.

Перечень поручений по итогам заседания Президиума Государственного Совета опубликован на официальном сайте Кремля.

«Правительству Российской Федерации совместно с комиссией Государственного Совета Российской Федерации по направлению «Образование» обеспечить наделение Министерства просвещения Российской Федерации правом устанавливать порядок экспертизы содержания и порядок лицензирования электронных образовательных

ресурсов, посредством которых обеспечивается реализация основных общеобразовательных программ и образовательных программ среднего профессионального образования, предусмотрев особенности регулирования использования электронных образовательных ресурсов при экспериментальном правовом режиме в сфере образования в соответствии с отдельными федеральными законами», — говорится в документе.

Срок исполнения поручения — до 1 января 2022 года.

Также правительству поручено обеспечить применение с 1 января 2023 года исключительно государственных информационных систем (ресурсов) при реализации основных общеобразовательных программ и образовательных программ среднего профессионального образования с использованием электронного обучения, дистанционных образовательных технологий, сопряженных с обработкой персональных данных обучающихся.

(По материалам федерального портала «Российское образование»)