

УПРАЖНЕНИЯ ПО ИНФОРМАТИКЕ КАК СРЕДСТВО ФОРМИРОВАНИЯ ОБЩЕУЧЕБНЫХ УМЕНИЙ ШКОЛЬНИКОВ

Exercises on computer science as means of developing student's universal learning activities

Атрощенко Светлана Аскольдовна, кандидат педагогических наук, доцент, Национальный исследовательский Нижегородский государственный университет им Н.И. Лобачевского Арзамасский филиал, Православный Свято-Тихоновский гуманитарный университет.

 atrochshenko_s@mail.ru

Первушкина Елена Александровна, кандидат педагогических наук, доцент, Национальный исследовательский Нижегородский государственный университет им Н.И. Лобачевского Арзамасский филиал.

 pervushkina@narod.ru

Для выполнения совокупности требований образовательных стандартов среднего образования, в которых заложен метапредметный подход к результатам обучения, необходимо решение проблемы организации учебной деятельности, ориентированной на сформированность у выпускников метапредметных умений. В данной статье под метапредметными умениями понимаются присвоенные обучающимися универсальные способы деятельности. В статье рассмотрены требования к упражнениям по информатике, направленным на формирование умения работать с информацией, освещены методические аспекты использования таких упражнений в практике обучения.

To fulfill a set of requirements of educational standards of secondary education, which laid metasubject approach to learning outcomes, it is necessary to solve the problem of organization of educational activities which aimed to develop of graduates' metasubject skills. In this article metasubject skills are considered as general learning abilities which were learned by students. The article describes the requirements for the exercise in computer science, aimed at the formation of the ability to work with information, shows methodological aspects of using of such exercises in the teaching practice.

Ключевые слова: **метапредметность, метапредметные умения, универсальные учебные действия, информация.**

Keywords: **metasubject, metasubject skills, universal learning activities, information.**

Современная российская школа должна удовлетворять требованиям Федеральных государственных образовательных стандартов (ФГОС), ориентированным на метапредметность, при которой на первый план в обучении выступают метапредметные умения, сформированные у выпускников.

Существуют разные трактовки понятия «метапредметность». Так, по мнению Н.В. Громыко, это «метапредметы», новая образовательная форма интегрирования учебного материала согласно «мыследеятельностному» типу [1]. А.В. Хуторской считает, что метапредметность - это «фун-

даментальные образовательные объекты» [2]. При описании метапредметного подхода в образовании исследователи используют различную терминологию: метапредметные умения, метапредметные компетентности, метапредметная деятельность, метапредметные результаты, метапредметные технологии. В данной статье, ориентируясь на терминологию ФГОС, под метапредметными умениями понимаются присвоенные обучающимися универсальные способы деятельности.

В условиях глобальной информатизации общества доминирующим видом деятельности в сфере общественного производства становится использование информации: ее сбор, накопление, продуцирование, обработка, хранение, передача. Как следствие, возникает серьезная проблема избыточности информации. Современному школьнику необходимо обладать умением приобретать, творчески интерпретировать информацию в обучении и будущей профессиональной деятельности.

Сказанное ставит проблему организации учебно-познавательной деятельности учащихся, способствующей формированию умения находить и перерабатывать нужную информацию с целью решения поставленных задач, как одного из важнейших метаумений.

Средством решения названной проблемы представляется использование в обучении специальных упражнений, удовлетворяющих требованиям:

- 1) развивать общеобразовательный уровень работы с информацией и готовить к продолжению обучения в вузе;
- 2) носить практическую направленность;
- 3) учитывать специфику содержания конкретной изучаемой дисциплины.

С учетом сформулированных требований были отобраны упражнения по информатике направленные на освоение действий, составляющих умение работать с текстом: операции, обеспечивающие понимание текста, его структурирование, выделение главного и второстепенного, основной идеи, выстраивание последовательности описываемых событий.

Рассмотрим пример упражнения по информатике для учащихся 11 класса.

Упражнение 1. Теория игр.

Условие: Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в одну из куч (по своему выбору) один камень или увеличить количество камней в куче в два раза. Например, пусть в одной куче 10 камней, а в другой 7 камней; такую позицию в игре будем обозначать $(10, 7)$. Тогда за один ход можно получить любую из четырёх позиций: $(11, 7)$, $(20, 7)$, $(10, 8)$, $(10, 14)$. Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 73. Победителем считается игрок,

сделавший последний ход, т.е. первым получивший такую позицию, что в кучах всего будет 73 камня или больше.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. Например, при начальных позициях (6, 34), (7, 33), (9, 32) выигрышная стратегия есть у Пети. Чтобы выиграть, ему достаточно удвоить количество камней во второй куче.

Задание 1. Для каждой из начальных позиций (6, 33), (8, 32) укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии.

Задание 2. Для каждой из начальных позиций (6, 32), (7, 32), (8, 31) укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии.

Задание 3. Для начальной позиции (7, 31) укажите, кто из игроков имеет выигрышную стратегию. Опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии. Постройте дерево всех партий, возможных при указанной Вами выигрышной стратегии. Представьте дерево в виде рисунка или таблицы [3].

Данное задание выполняется последовательно. Решив первую часть задания, мы легко можем решить следующую.

Задание 1. Кто из игроков имеет выигрышную стратегию при позициях (6, 33) и (8, 32)?

Нецелесообразно начинать решение задачи с построения таблицы поочередных ходов каждого из игроков, так как в случае исходных значений по нескольким начальным позициям это будет сводиться к выполнению математических операций по подсчету разнообразных вариантов ходов и получения числа камней в каждой из куч у определенного игрока на каждом шаге игры. Нам же необходимо научить анализировать исходную ситуацию и моделировать процесс игры с учетом всевозможных ходов игрока-противника. В связи с этим будем придерживаться аналитического построения рассуждений в процессе выделения необходимой информации из условия задачи. Читаем задание и выделяем ключевую мысль в тексте и в решении задачи.

Игроки могут делать два хода — прибавить к любой из куч один камень, или умножить количество камней в любой из куч на два. Игрок выигрывает, когда суммарное количество камней в куче становится ≥ 73 .

Чтобы продолжить рассуждения, введем два необходимых понятия. Будем считать, что прибавление одного камня в маленькую кучу — *самый слабый ход*, а увеличение в два раза большой кучи — *самый сильный ход*. Остальные нам не интересны, так как такие ходы уже покажут положение игрока в данной ситуации. Первый ход делает Петя. Рассмотрим это на графе (рис. 1):



Рис. 1. Ходы первого игрока по стратегии

Мы видим, что Петя первым ходом не может получить суммарное количество ≥ 73 , зато Ваня сможет это сделать следующим своим ходом, удвоив количество камней во второй куче:

$$\begin{aligned} 7 + 2*33 &= 73 \\ 6 + 2*66 &= 138 \\ 9 + 2*32 &= 73 \\ 8 + 2*64 &= 136 \end{aligned}$$

Таким образом, при позициях (6, 33) и (8, 32) второй игрок (Ваня) выигрывает своим первым ходом. Так как Петя, получив такие позиции, проиграл, будем считать, что позиции (6, 33) и (8, 32) проигрышные, и игрок, которому они достанутся, проиграет.

Этот вывод мы будем использовать при дальнейшем решении задачи.

Задание 2. Кто из игроков имеет выигрышную стратегию при позициях (6, 32), (7, 32), (8, 31)?

Рассуждаем аналогичным образом. Петя ходит первым, значит, чтобы выиграть, он должен сделать так, чтобы Ване досталась проигрышная позиция. Мы знаем две такие проигрышные позиции из пункта 1 - (6, 33) и (8, 32). Их Петя может сделать своим первым ходом для Вани из предложенных позиций прибавлением одного камня в одну из куч:

$$\begin{aligned} (6, \underline{32+1}) &\rightarrow (6, 33) \\ (\underline{7+1}, 32) &\rightarrow (8, 32) \\ (8, \underline{31+1}) &\rightarrow (8, 32) \end{aligned}$$

Таким образом, Петя сделал для Вани проигрышную позицию, вследствие чего Ваня проиграет, а Петя выиграет своим вторым ходом. Поэтому, будем считать, что позиции (6, 32), (7, 32), (8, 31) выигрышные, так как они достались Пете и Петя выиграл.

Аналогичным образом решается третье задание и формулируется ответ к задаче.

Упражнение 2. Разработка программы.

Условие: Специальная камера, установленная на перекрёстке, фиксирует количество проезжающих автомобилей, и каждую минуту по каналу связи передаёт неотрицательное целое число — количество автомобилей, проехавших перекрёсток за эту минуту. Известно, что за минуту перекрёсток может проехать не более 100 автомобилей. Необходимо найти в заданной серии показаний максимальное количество автомобилей, проехавших перекрёсток в течение пяти подряд идущих минут. Максимальное количество показаний, которое может передать камера, не превышает 1440.

Напишите на любом языке программирования программу для решения поставленной задачи. Для получения максимального результата программа должна быть эффективна по времени и по используемой памяти. Входные данные представлены следующим образом (см. Таблица 1). В первой строке задаётся число N – общее количество переданных показаний. Гарантируется, что $N > 5$. В каждой из следующих N строк задаётся одно положительное целое число – очередное показание камеры [3].

Пример входных и выходных данных программы

<i>Пример входных данных:</i>	8 5 12 27 10 4 50 7 16
Программа выводит только одно число – наибольшее количество автомобилей, проехавших перекресток за пять подряд идущих минут.	
<i>Пример выходных данных для приведённого выше примера входных данных:</i>	103

Выделим основные моменты в решении этой задачи [4].

1. Правильный алгоритм.

До того, как начать программировать, необходимо тщательно продумать алгоритм, учесть все возможные варианты поведения программы. После этого обязательно подробно и понятно записать алгоритм на простом языке, в виде блок-схемы или в виде таблицы.

2. Эффективность.

В постановке задачи требуется написать эффективную программу, что означает продемонстрировать умение экономно расходовать основные ресурсы: память компьютера и время. Смысл задачи – проверить умение распоряжаться ограниченными ресурсами.

А) Эффективность по времени.

Наиболее ценным ресурсом в этой задаче считается время. Эффективность по времени имеет преимущество перед эффективностью по памяти. Чтобы написать эффективную по времени программу, нужно стараться избегать вложенных циклов.

Б) Эффективность по памяти.

Все, что выполняет программа, происходит в памяти компьютера. Объявляя переменные, мы резервируем ячейки памяти (переменная типа Integer занимает в классическом Паскале 2 байта, переменная типа Real – 6 байт). Записывая введенные данные в массив или переменные, мы используем память. Поэтому основные приемы экономии памяти: а) правильно выбирать тип переменной; б) при возможности не сохранять вводимые данные в массив или переменные, а анализировать сразу при вводе; в) экономно использовать переменные (если возможно, использовать одну переменную для разных целей). После написания программы необходимо сделать анализ эффективности. Объяснить, почему были выбраны такие типы переменных. Указать, в какой именно части программы экономно была использована одна и та же переменная в разных целях [4].

3. Правила оформления программного кода.

Рекомендуется выполнять следующие правила: а) целесообразно использовать имена переменных, указывающие на их назначение; б) желательно форматировать текст отступами, обозначая начало-конец программных блоков, что избавляет от потери закрывающих скобок и упростит чтение текста; в) необходимо использовать комментарии, коротко описывающие основной смысл происходящего [4].

В Таблице 1 представлен образец оформления решения задачи.

Таблица 1.

Образец оформления решения задачи

Аналитическое описание	Программный код	Полное решение
Для решения задания можно не использовать такой огромный массив, достаточно хранить в памяти всего пять значений, сумму которых необходимо найти:	var a: array [1..5] of byte;	var a: array [1..5] of integer; s, i, j, max, N: integer; begin readln(n); s := 0;
Сначала введем N и считаем в массив первые пять значений, при этом сразу можно найти и их сумму:	readln(n); for i:=1 to 5 do begin readln(a[i]); s := s+a[i]; end ;	for i:=1 to 5 do begin readln(a[i]); s := s+a[i]; end ;
Сразу же присвоим переменной <i>max</i> значение <i>s</i> :	max := s;	max := s;
Теперь введём оставшиеся зна-	for i:=6 to N do	for i:=6 to N do begin s := s — a[i];

<p>чения. Для этого будем использовать последний (пятый) элемент массива, предварительно сдвинув все элементы на один влево:</p>	<pre>begin for j:=1 to 4 do //сдвиг массива a[j] := a[j+1]; readln(a[5]);</pre>	<pre>for j:=1 to 4 do a[j] := a[j+1]; readln(a[5]); s := s + a[5]; if s > max then max :=s; end; writeln(max); end.</pre>
<p>Можно посчитать сумму элементов с помощью вложенного цикла, но можно поступить и следующим образом: сумму элементов можем посчитать, вычтя из предыдущей суммы первый элемент до сдвига, и прибавив к ней введённый элемент после сдвига:</p>	<p>То есть теперь код будет выглядеть так: for i:=6 to N do</p> <pre>begin s := s — a[1]; for j:=1 to 4 do a[j] := a[j+1]; readln(a[5]); s := s + a[5]</pre>	
<p>Остаётся сравнить значение переменной <i>s</i> с <i>max</i>:</p>	<pre>if s > max then max :=s;</pre>	

Результатом аналитического исследования условия задачи и полученного решения в процессе выполнения упражнений учащимися, является развитие одного из важнейших метаумений: находить и перерабатывать нужную информацию, которую можно использовать в дальнейшей жизнедеятельности [5].

Предложенные требования к упражнениям по информатике являются вариативными и могут быть адаптированы к любой школьной дисциплине.



БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Громыко Н. В. Метапредметный подход в образовании: как сценарировать и проводить учебное «метапредметное» занятие, реализуя новые образовательные стандарты // НИИ инновационных стратегий развития общего образования: Вестник 2010-2011. -М.: НИИ ИСРОО, Пушкинский институт, 2010 -2011. – С. 114 -119.
2. Хуторской А.В. Метапредметный подход в обучении: Научно-методическое пособие / А.В. Хуторской. М.: Эйдос; Изд-во Института образования человека, 2012. – С. 50.
3. Информатик БУ. Подготовка к ЕГЭ-2016 по информатике и ИКТ - Режим доступа: <http://infbu.ru/>
4. Есакова Л.Б. Самая сложная задача на ЕГЭ по информатике (задача 27, 4 первичных балла). - Режим доступа:<http://ege-study.ru/ege-informatika/zadacha-27-na-ege-po-informatike>.
5. Атрощенко С.А., Первушкина Е.А. Использование моделирующих компьютерных программ в профессионально направленном обучении геометрии // Международный научно-исследовательский журнал. 2014. № 7-2 (26). С. 47-49.