**УДК** **373.3,** **373.5** **ББК** **74.26**

***Бесшапошников*** ***Никита*** ***Олегович,*** ***младший*** ***научный*** ***сотрудник;*** ***Кушниренко*** ***Анатолий*** ***Георгиевич,*** ***к.ф.-м.н.,*** ***доцент,*** ***зав.*** ***отделом;*** ***Малый*** ***Александр*** ***Альбертович,*** ***младший*** ***научный*** ***сотрудник;*** ***Прилипко*** ***Алексей*** ***Алексеевич,*** ***к.ф.-м.н.,*** ***зам.*** ***директора*** ***по*** ***научной*** ***и*** ***методической*** ***работе,***

***Федеральное*** ***Государственное*** ***Учреждение*** ***«Федеральный*** ***Научный*** ***Центр*** ***Научно-исследовательский*** ***институт*** ***системных*** ***исследований*** ***Российской*** ***Академии*** ***наук»*** ***(ФГУ*** ***ФНЦ*** ***НИИСИ*** ***РАН),*** ***г.*** ***Москва***

ТРИ ЦИКЛА КОМБИНАТОРНЫХ ЗАДАЧ ПО ПРОГРАММИРОВАНИЮ ДЛЯ СТАРШЕКЛАССНИКОВ С ИСПОЛЬЗОВАНИЕМ ГЕОМЕТРИЧЕСКИХ СРЕД «КВАДРАТИЯ» И «КУБЕРИЯ»

***Аннотация:*** *В* *статье* *обосновывается* *необходимость* *выхода* *за* *рамки* *«программистских»* *миров* *при* *вве-дении* *и* *освоении* *понятия* *многомерный* *массив.* *Предлагается* *на* *начальном* *этапе* *изучения* *многомерных* *массивов* *использовать* *простейшие* *геометрические* *объекты* *на* *плоскости* *и* *пространстве.* *Определяются* *предметные* *об-ласти* *Квадратия* *и* *Куберия* *и* *приводятся* *три* *цикла* *геометрических* *задач,* *решаемых* *в* *этих* *предметных* *областях.* *В* *приложении* *к* *статье* *разобраны* *решение* *элементарной* *задачи* *и* *задачи* *среднего* *уровня* *сложности.*

***Ключевые*** ***слова:*** *школьная* *информатика,* *многомерный* *массив,* *алгоритмы* *работы* *с* *массивами,* *геометри-ческий* *объект,* *клетка,* *фигура,* *граница,* *непустота,* *связность,* *квадрат,* *куб,* *целочисленные* *координаты,* *алгоритм,* *Квадратия,* *Куберия,* *робот* *Вертун,* *система* *КуМир.*

**1.** **Введение.** Свободное владение элементарными приемами работы с многомерными массивами – один из плани-руемых результатов изучения углубленного курса информатики старшей школы. Это свободное владение может быть достигнуто только путем решения нескольких десятков задач различной степени сложности. Несмотря на достаточно развитый уровень логического и абстрактного мышления старшеклассников, избравших изучение ин-форматики на углубленном уровне, выработка навыков работы с массивами исключительно на задачах, сформу-лированных в чисто «программистских» терминах, нередко приводит к непониманию старшеклассниками сути и важности решаемых задач. Процесс обучения станет более эффективным, если на начальном этапе освоения сами массивы и задачи анализа содержащейся в них информации окажутся не предметом изучения, а лишь инструмен-том решения задач в предметных областях, лежащей вне программирования. Теоретически, одной из таких пред-метных областей могла бы стать физика. Практически, такой подход мог бы применяться только при условии фактической и формальной интеграции учебных программ физики и информатики, которая в настоящий момент осуществима только в системе дополнительного образования.

**2.** **Предложения** **по** **выбору** **предметной** **области.** Предлагается, не выходя за пределы учебной программы инфор-матики, в качестве полигона для начальной отработки техники работы с массивами использовать геометрию про-стейших геометрических объектов в двумерном и трехмерном пространствах. Такие объекты легко визуализиру-ются и при работе с ними можно опираться на накопленный обучаемыми богатый чувственный опыт манипуля-ции объектами в трехмерном пространстве.

**3.** **Описание** **выбранных** **геометрических** **сред.** Для того, чтобы при работе в этой предметной области не потребо-вались математические знания, выходящие за пределы школьной программы (в первую очередь, знание аналити-ческой геометрия), в качестве геометрических объектов можно взять конечные объединения единичных квадратов на плоскости и единичных кубов в трехмерном пространстве с целыми координатами вершин. Далее такие еди-ничные квадраты и кубы будут называться ***клетками***. А составленные из клеток объекты будут называться **фи-гурами**. Использование терминов ***клетка*** и ***фигура*** позволит избежать упоминания размерности геометрического объекта в тех случаях, когда размерность не важна.

**4.** **Свойства** **и** **характеристик** **фигур.** Класс фигур замкнут относительно операций объединения и пересечения. Для фигур имеет смысл большинство определений свойств и числовых характеристик двумерных областей и трехмерных тел. В частности, можно говорить о **непустоте** фигуры, **гладкости** фигуры, **границе** фигуры, **связно-сти** фигуры **и** **числе** **ее** **связных** **компонент**, **объеме** и **площади** **поверхности** **границы** фигуры (трехмерный случай) и **площади** и **периметре** фигуры (двумерный случай).

**5.** **Дополнительные** **возможности** **при** **работе** **с** **предложенными** **геометрическими** **средами.** Введение таких сред позволяет проиллюстрировать и отработать понятие рекурсии. В классах с углубленной математической подго-товкой задачи, могут быть предложены задачи по программированию, постановка или алгоритм решения которых опираются на дополнительный математический материал. Такова, например, задача о нахождении рода (числа «ручек») двумерной поверхности.

**6.** **Программная** **реализация.** Для визуализации фигур и манипуляций с ними был разработан модуль визуализации трехмерных фигур **«Куберия».** Модуль проинтегрирован в качестве исполнителя в учебную систему программи-рования КуМир. Параллельно был разработан и включен в КуМир исполнитель **«Квадратия».** Квадратия и Кубе-рия ориентированы на углубленный курс информатики старших классов и систему дополнительного образования, но эпизодически могут быть использованы и в базовом курсе информатики основной школы. Задачи, пригодные для такого использования, могут быть найдены в п. 10.

**7.** **Связь** **с** **методиками** **начальной** **школы.** Напрашивающееся распространение предложенного подхода на одно-мерный случай оказывается малоосмысленным: для исполнителя с условным названием «Одномерия» не нашлось бы достаточного количества содержательных задач. Однако, если вместо одной одномерной прямой рассмотреть сеть пересекающихся горизонтальных и вертикальных прямых на плоскости (мир клетчатой бумаги), то получит-ся содержательная обучающая среда для младшеклассников, описанная в статье М. Ройтберга [1]. Исполнитель «Клеточки» для поддержки этой среды, дополняющий ряд «Куберия» и «Квадратия», планируется разработать и проинтегрировать в КуМир в 2017 году.

**8.** **Особенности** **систем** **команд** **исполнителей** **Куберия** **и** **Квадратия.** Получение информации о фигуре в Квадра-тии и Куберии возможно в одном из двух взаимоисключающих режимов.

**Режим** **«дальнодействия».** В этом режиме на множестве всех клеток вводится целочисленная система коор-динат: x1, x2 – на плоскости; x1, x2, x3 – пространстве, и фигура задается информацией о том, какие клетки ей при-надлежат. Эту информацию поставляет логическая функция **клетка**, выдающее логическое значение **да**, если в каче-стве аргументов указаны координаты клетки, принадлежащей фигуре, и выдающей значение **нет** в противном случае.

По определению, фигура конечна. Если фигура непуста, содержит хотя бы одну клетку, то такая клетка может быть найдена за конечное число шагов. Однако проверить пуста фигура или нет, не располагая какой-то априорной информацией о расположении фигуры, за конечное число шагов невозможно.

Поэтому постулируется, что фигуры могут располагаться только в некоторой заранее заданной области, назы-ваемой ***Поле***. Исполнители Квадратия и Куберия в режиме дальнодействия предоставляют априорную информацию о границах Поля и числе клеток Поля.

|  |  |
| --- | --- |
| Квадратия | Куберия |
| **лог** **клетка** (**цел** x, **цел** у) | **лог** **клетка**(**цел** x, **цел** у, **цел** z) |
| **цел** **X1min** **цел** **X1max** **цел** **X2min** **цел** **X2max** | **цел** **X1min** **цел** **X1max** **цел** **X2min** **цел** **X2max** **цел** **X3min** **цел** **X3max** |
| **цел** **Число** **Клеток** **Поля** | |

**Замечание.** Однако функция **клетка** определена для любых аргументов, а не только для аргументов из Поля.

С использованием априорной информации о Поле, определение пустоты/непустоты фигуры и поиск клетки, принад-лежащей непустой фигуре, становятся выполнимыми за конечное число шагов.

**Режим** **«близкодействия**». В этом бескоординатном режиме на поле Квадратии или Куберии располагается агент – робот ***Вертун*.** Вертун умещается в одной клетке, ориентирован и по команде

может сместиться на один шаг в соседнюю клетку, если при этом он не пересекает границу фигуры. Атомарным эле-ментом границы является единичный отрезок в случае Квадратии и единичный квадрат в случае Куберии. В режиме близкодействия элементы границы фигуры называются **стенами.** Информацию о фигуре (точнее, о ее границе) можно получить с помощью двух команд-вопросов Вертуна (логических функций без аргументов):

**лог** **впереди** **стена** **лог** **впереди** **свободно**

Изменение ориентации Вертуна выполняется командами

|  |  |  |  |
| --- | --- | --- | --- |
| Квадратия | Куберия | | |
| **повернуть** **направо** **повернуть** **налево** | **повернуть** **направо** **повернуть** **налево** | **штурвал** **на** **себя** **штурвал** **от** **себя** | **по** **часовой** **против** **часовой** |

Команды изменения ориентации не меняют клетки, в которой находится Вертун, и являются ***относительными*** – ре-зультат выполнения команды зависит от ориентации на начало выполнения команды. В трехмерном случае, команды поворота **по** **часовой** и **против** **часовой** не меняют и клетку, на которую направлен Вертун. Имеется ***абсолютная*** команда

**восстановить** **ориентацию**

выполнение которой придает Вертуну некоторую ***стандартную*** ***ориентацию***, независимо от его текущей ориента-ции.

Задачи для исполнителей Квадратия и Куберия делятся на три группы, три цикла.

**9.** **Первый** **цикл** **задач.** Здесь предлагаются бесхитростные задачи, не требующие для своего решения сложных ал-горитмов и почти не требующие математических рассуждений. Эти задачи могут эпизодически решаться и в базо-вом курсе информатики.

**Пояснение.** Если в формулировке задачи не указана размерность, это указывает на то, что задача решается и на плос-кости и в пространстве более или менее одинаковым образом.

**Задача** **1.** *Определить,* *принадлежит* *ли* *фигуре* *хотя* *бы* *одна* *клетка.*

**Задача** **2.** *Дана* *непустая* *фигура.* *Найти* *координаты* *какой-нибудь* *клетки* *этой* *фигуры.* **Задача** **3.** *Найти* *число* *клеток* *фигуры.*

**Задача** **4.** *Найти* *длину* *границы* *(периметр)* *плоской* *фигуры.* **Задача** **5.** *Найти* *площадь* *границы* *трехмерной* *фигуры.*

|  |
| --- |
| **Мини-теория** **ящиков.** |
| ***Ящиком*** *называется* *фигура,* *составленная* *изо* *всех* *клеток,* *координаты* *которых* *удовлетворяют* *неравен-ствам*  Разности bi-ai называются ***длинами*** ***ребер*** ящика.  Ящик **непуст** тогда и только тогда длины всех его ребер положительны. Число клеток в непустом ящике равно произведению длин его ребер. Пересечение двух ящиков является ящиком.  Для каждой непустой фигуры существует и единственен содержащий ее ящик с минимальным числом кле-ток.  Непустой ящик называется ***правильным***, если длины всех его ребер равны. Правильный ящик в Квадратии называется квадратом, а в Куберии – кубом. **Пример** **1.** *Непустая* *клетка* *является* *ящиком,* *все* *длины* *сторон* *которого* *равны* *1.*  **Пример** **2.** *Множество* *всех* *клеток,* *содержащих* *вершину* *какой-нибудь* *клетки* *в* *Квадратии,* *содержит* *4* *клетки* *и* *является* *квадратом* *с* *длиной* *ребра* *2.*  **Пример** **3.** *Множество* *всех* *клеток,* *содержащих* *вершину* *какой-нибудь* *клетки* *в* *Куберии,* *содержит* *8* *клеток* *и* *является* *кубом* *с* *длиной* *ребра* *2.* |

**Задача** **6.** *Известно,* *что* *фигура* *непуста* *и* *является* *ящиком.* *Даны* *координаты* *одной* *из* *клеток* *этого* *ящика.* *Най-ти* *ребра* *ящика.*

**Задача** **7.** *Известно,* *что* *фигура* *непуста* *и* *является* *ящиком.* *Найти* *координаты* *какого-нибудь* *из* *углов* *этого* *ящи-ка.*

**Задача** **8.** *Дана* *непустая* *фигура.* *Определить,* *является* *ли* *она* *ящиком.*

**Задача** **9.** *Дана* *непустая* *фигура.* *Найти* *содержащий* *ее* *ящик* *с* *минимальным* *числом* *клеток.*

**Задача** **10.** *Известно,* *что* *фигура* *является* *объединением* *двух* *квадратов* *одинакового* *размера.* *Определить* *длину* *ребра* *этих* *квадратов.*

**Задача** **11.** *Известно,* *что* *фигура* *является* *объединением* *двух* *кубов* *одинакового* *размера.* *Определить* *длину* *ребра* *этих* *кубов.*

**Задача** **12.** *Дана* *фигура* *на* *плоскости.* *Найти* *ее* *периметр.*

**Задача** **13.** *Дана* *фигура* *в* *пространстве.* *Найти* *площадь* *поверхности* *ее* *границы.*

**Задача** **14.** *Вертун* *находится* *внутри* *некоторого* *ящика.* *Переместить* *Вертуна* *в* *один* *из* *углов* *ящика.*

**Задача** **15.** *Вертун* *находится* *внутри* *ящика* *с* *нечетными* *длинами* *ребер.* *Переместить* *Вертуна* *в* *центр* *симметрии* *ящика.*

**Задача** **16.** *На* *некотором* *расстоянии* *спереди* *или* *сзади* *Вертуна* *на* *плоскости* *находится* *стена.* *Подвести* *вертуна* *к* *стене.*

**Задача** **17.** *На* *некотором* *расстоянии* *спереди* *или* *сзади,* *слева* *или* *справа* *от* *Вертуна* *на* *плоскости* *находится* *сте-на.* *Подвести* *Вертуна* *к* *стене.*

**Задача** **18.** *Вертун* *находится* *внутри* *ящика* *на* *плоскости.* *Закрасить* *все* *стены* *ящика* *изнутри.* **Задача** **19.** *Вертун* *находится* *внутри* *ящика* *на* *плоскости.* *Закрасить* *все* *клетки* *ящика.*

**Задача** **20.** *Вертун* *находится* *внутри* *ящика* *на* *плоскости.* *Закрасить* *все* *клетки* *ящика,* *прилегающие* *к* *его* *границе.* **Задача** **21.** *Вертун* *находится* *вне* *ящика* *на* *плоскости.* *Впереди* *Вертуна* *стена.* *Закрасить* *все* *стены* *ящика* *снару-жи.*

**Задача** **22.** *Вертун* *находится* *внутри* *ящика* *в* *пространстве.* *Закрасить* *все* *клетки* *ящика.*

**10.** **Необходимость** **обогащения** **внутренней** **структуры** **исполнителей** **Квадратия** **и** **Куберия** **для** **более** **ком-фортного** **решения** **задач** **циклов** **2** **и** **3.**

Для решения задач циклов 2 и 3 одной информации о расположении фигуры на поле недостаточно. В процессе реше-ния многих из этих задач требуется нагружать элементы фигуры дополнительной информацией. Чтобы избавить школьника от необходимости определять дополнительные многомерные массивы в своей программе, исполнители Квадратия и Куберия позволяют задавать и считывать некоторую дополнительную информацию, сохраняя ее внутри исполнителя (помечать клетки и границы клеток). Эта дополнительная информация о фигуре жизненно необходима для реализации всех рекурсивных алгоритмов, а также может быть использована и в других случаях.

|  |
| --- |
| **Мини-теория** **пометок** **клеток** **и** **их** **границ** **в** **режиме** **близкодействия** |
| В этом режиме доступны команды-приказы **закрасить** **клетку**  **закрасить** **стену**  позволяющие отметить клетку, в которой Вертун находится и элемент границы, на который Вертун на-правлен. Эти команды не имеют аргументов.  Для получения информацию о сделанных пометках служат команды-вопросы **лог** **клетка** **закрашена**  **лог** **клетка** **чистая** **лог** **стена** **закрашена** **лог** **стена** **чистая**  При попытке закрасить стену или получить информацию о закраске стены в ситуации, когда **впереди** **свободно**, возникает отказ. |

**Режим** **дальнодействия.** В этом режиме имеется возможность нагрузки клеток и их границ целочисленной информа-цией. Благодаря погружению этой дополнительной информации «внутрь» исполнителей Квадратия и Куберия, рекур-сивные алгоритмы учеников упрощаются начинают умещаться на один экран.

**Определение.** Клетки Поля называются соседними, если у них совпадают значения всех координат, кроме одной, зна-чение которой отличается на +1 или -1.

Упорядоченная пара соседних клеток, одна из которых принадлежит фигуре, а другая нет, определяют элемент грани-цы клетки (единичный отрезок в двумерном случае и единичный квадрат в трехмерном). Пометить этот элемент гра-ницы можно как с одной стороны, так и с другой или одновременно с двух сторон.

|  |
| --- |
| **Мини-теория** **пометок** **клеток** **и** **их** **границ** **в** **режиме** **дальнодействия** |
| В этом режиме процедуры исполнителей Квадратия и Куберия, позволяют помечать клетки По-ля и границы клеток Поля целыми числами и читать сделанные пометки. При инициализации исполни-теля все пометки определены и равны нулю.  **Квадратия:**  **пометить** **клетку** **(цел** **n,** **x1,** **x2)**  **пометить** **границу** **(цел** **n,** **x1,** **x2,** **dx1,** **dx2)** **цел** **пометка** **клетки** **(цел** **x1,** **x2)**  **цел** **пометка** **границы** **(цел** **x1,** **x2,** **dx1,** **dx2)**  **где** **одна** **компонента** **вектора** **(dx1,** **dx2)** **должны** **быть** **равна** **нулю,** **а** **другая** **–** **равна** **+1** **или** **-1.**  **Куберия**  **пометить** **клетку** **(цел** **n,** **x1,** **x2,** **x3)**  **пометить** **границу** **(цел** **n,** **x1,** **x2,** **x3,** **dx1,** **dx2,** **dx3)** **цел** **пометка** **клетки** **(цел** **x1,** **x2)**  **цел** **пометка** **границы** **(цел** **x1,** **x2,** **dx1,** **dx2)**  **где** **две** **компоненты** **вектора** **(dx1,** **dx2,** **dx3)** **должны** **быть** **равны** **нулю,** **а** **третья** **–** **равна** **+1** **или** **-1.** |

**11.** **Второй** **цикл** **задач.** В этом цикле предлагаются более сложные задачи, требующие для своего решения изобрете-ния не вполне тривиальных алгоритмов, однако не требующие дополнительных математических сведений.

**Пояснение.** Если в формулировке задачи не указана размерность, это указывает на то, что задача решается и на плос-кости и в пространстве более или менее одинаковым образом.

**Задача** **1.** *На* *поле* *задана* *непустая* *фигура,* *количество* *клеток* *в* *которой* *составляет* *не* *менее* *половины* *величины* Число Клеток Поля. *Найти* *координаты* *какой-нибудь* *клетки* *фигуры* *методом* *Монте-Карло.*

**Задача** **2.** *На* *поле* *задана* *непустая* *фигура,* *количество* *клеток* *в* *которой* *составляет* *не* *менее* *половины* *величины* Число Клеток Поля. *Найти* *приближенно* *число* *клеток* *фигуры* *методом* *Монте-Карло.*

**Задача** **3.** *Даны* *непустая* *связная* *фигура* *и* *координаты* *некоторой* *клетки* *фигуры.* *Закрасить* *все* *клетки* *фигуры* *и* *найти* *число* *клеток* *фигуры* *с* *помощью* *рекурсивного* *обхода* *всех* *клеток* *фигуры.*

|  |
| --- |
| **Мини-теория** **связности** **фигур** **и** **их** **границ** |
| **Определение.** *Фигура* *называется* ***связной***, *если* *помещенный* *в* *одну* *из* *клеток* *фигуры* *Вертун,* *может* *добраться* *до* *любой* *другой* *клетки* *фигуры.*  **Определение.** *Множество* *точек* *фигуры,* *до* *которых* *Вертун* *может* *добраться,* *стартуя* *из* *некото-рой* *точки,* *называется* **связной** **компонентой**. *Если* *фигура* *связна,* *то* *она* *состоит* *из* *одной* *компо-ненты* *связности.* *Если* *фигура* *не* *связна,* *то* *число* *ее* *связных* *компонент* *больше* *1.*  **Определение.** *Граница* *фигуры* *называется* ***связной***, *если* *перемещаясь* *по* *ребрам* *границы,* *можно* *доб-раться* *из* *любой* *вершины* *границы* *в* *любую* *другую.*  **Пример** **1.** *Каждый* *непустой* *ящик* *является* *связной* *фигурой.*  **Пример** **2.** *Объединение* *двух* *клеток,* *имеющих* *только* *одну* *общую* *точку,* *не* *является* *связной* *фигурой* *и* *состоит* *из* *двух* *связных* *компонент.* *Тем* *не* *менее,* *граница* *этой* *фигуры* *является* *связной.*  **Пример** **3.** *Объединение* *двух* *клеток* *в* *пространстве,* *имеющих* *общее* *ребро,* *не* *является* *связной* *фигу-рой* *и* *состоит* *из* *двух* *связных* *компонент.*  **Пример** **4.** *Квадрат* *размером* *3х3,* *из* *которого* *удален* *центральный* *квадратик* *размером* *1х1,* *является* *связной* *фигурой.* *Граница* *этой* *фигуры,* *однако,* *связной* *не* *является* *и* *состоит* *из* *двух* *компонент* *свзности.* |

Задачи 4-6 должны решаться в режиме дальнодействия, без использования Вертуна, с помощью рекурсивных алго-ритмов.

**Задача** **4.** *Дана* *непустая* *фигура.* *Определить,* *связна* *ли* *эта* *фигура.* **Задача** **5.** *Дана* *непустая* *фигура.* *Определить* *число* *ее* *связных* *компонент.*

**Задача** **6.** *Дана* *непустая* *фигура.* *Найти* *максимальное* *среди* *чисел* *клеток* *в* *ее* *связных* *компонентах.*

|  |
| --- |
| ***Мини-теория*** ***гладкости*** ***фигур,*** ***составленных*** ***из*** ***клеток*** |
| **Определение.** *Граница* *фигуры* *на* *плоскости* *называется* ***гладкой****,* *если* *для* *любой* *вершины* *границы* *пересече-ние* *границы* *с* *внутренностью* *квадрата* *2х2* *с* *центром* *в* *этой* *вершине* *представляет* *собой* *объединение* *двух* *полуинтервалов,* *пересекающихся* *в* *центре* *квадрата* *2х2.*  **Определение.** *Граница* *фигуры* *в* *пространстве* *называется* ***гладкой****,* *если* *для* *любой* *вершины* *границы* *пересе-чение* *границы* *с* *внутренностью* *куба* *2х2х2* *с* *центром* *в* *этой* *вершине* *представляет* *собой* *объединение* *3,* *4,* *5* *или* *6* *квадратов,* *которые* *пересекаются* *в* *центре* *куба* *2х2х2* *и* *могут* *быть* *циклически* *упорядочены* *так,* *что-бы* *каждый* *квадрат* *пересекался* *со* *следующим* *по* *ребру,* *исходящему* *из* *центра* *куба* *2х2х2.* |

**Задача** **7.** *Дана* *непустая* *фигура* *на* *плоскости.* *Определить,* *гладка* *ли* *ее* *граница*.

В задачах 8-10 дана плоская связная фигура со связной гладкой границей. Вертун стоит вплотную к стене этой границы. В процессе выполнения алгоритма Вертун не должен удаляться от границы.

**Задача** **8.** *Определить* *длину* *границы* *(периметр* *фигуры).* **Задача** **9.** *Определить* *площадь* *фигуры.*

**Задача** **10.** *Определить,* *находится* *Вертун* *внутри* *или* *снаружи* *фигуры.*

**Задача** **11.** *Выяснить,* *может* *ли* *Вертун* *добраться* *до* *границы* *заданной* *фигуры* *не* *более* *чем* *за* *10* *перемещений.* **Задача** **12.** *Подвести* *Вертуна* *к* *границе* *заданной* *непустой* *фигуры.*

**12.** **Третий** **цикл** **задач**. В этом цикле предлагаются трудные задачи, требующие для своего решения математических рассуждений и приемов.

**Задача** **1.** *Дана* *непустая* *фигура.* *Выяснить,* *гладка* *ли* *ее* *граница.*

**Задача** **2.** *Есть* *255* *непустых* *фигур* *в* *пространстве,* *умещающихся* *в* *куб* *размером* *2x2x2.* *Подсчитать,* *сколько* *из* *этих* *фигур* *имеют* *негладкую* *границу.*

В задачах 3-6 дана трехмерная связная фигура со связной гладкой границей. Вертун стоит вплотную к стене этой границы. В процессе выполнения алгоритма Вертун не должен удаляться от границы.

**Задача** **3.** *Определить* *площадь* *границы* *фигуры.* **Задача** **4.** *Определить* *объем* *фигуры.*

**Задача** **5.** *Определить,* *находится* *Вертун* *внутри* *или* *снаружи* *этой* *фигуры.*

**Задача** **6.** *Выясните* *в* *интернете,* *как* *определяется* *число* *«ручек»* *трехмерной* *связной* *фигуры* *со* *связной* *гладкой* *границей.* *Определите* *число* *ручек* *фигуры.*

**Задача** **7.** *Определить,* *связна* *ли* *граница* *заданной* *плоской* *связной* *фигуры* *с* *гладкой* *границей.*

**Задача** **8.** *Определить,* *связна* *ли* *граница* *заданной* *пространственной* *связной* *фигуры* *с* *гладкой* *границей.*

**Дополнительная** **задача** **цикла** **3.** **Патрулирование** **шахматной** **доски.**

Вертун патрулирует шахматную доску (квадрат со стороной n=8), последовательно выполняя обходы всех клеток дос-ки, стартуя в левом нижнем углу доски в горизонтальном направлении. Поскольку Вертун наиболее уязвим в моменты поворота, он делает каждый обход с минимальным числом поворотов, которое, как можно доказать, равно 16. Чтобы сбить с толку злоумышленников, Вертун выполняет последовательно все возможные обходы.

**Задача** **9.** *Найти* *число* *различных* *обходов* *с* *минимальным* *числом* *поворотов.* *Проверить* *решение* *для* *n=4* *путем* *по-строения* *всех* *обходов* *с* *8* *поворотами* *вручную.*

**13.** **Программная** **реализация** **визуализатора** **Куберия.**

Куберия реализована с помощью кроссплатформенной библиотеки для разработки игр cocos2d-x [2]. Данная библиотека позволяет компилировать и запускать игру на нескольких платформах (Windows 7+, Mac OS, iOs, Android, Windows Phone) без написания дополнительного кода для их поддержки. Для реализации Куберии используется отно-сительно недавно добавленная поддержка 3D в cocos2d-x.

В качестве основы сцены с кубиками используется стандартная модель кубика в формате Wavefront OBJ и простая текстура для граней куба [3]. Кубы рисуются через стандартный класс cocos2d-x – *Sprite3D.* Они маштабиру-ются для соответствия размеру экрана с помощью матрицы для преобразования локальных координат модели. Кубы располагаются по трехмерной сетке, которая имеет координаты (0, 0, 0) в центре экрана и ось Z направленную на пользователя. Программа считывает из входного файла обстановки тройки целых чисел, которые отвечают за их рас-положение в этой сетке.

В модуле реализована поддержка камеры типа “*arcball*”. Данный тип камеры позволяет вращать, приближать и отдалять сцену по дугам сферы, расположенной в центре нее. Вращение обстановки происходит с помощью движе-ний мышки или пальца на мобильных устройствах. Приближение и отдаление работает по нажатию кнопок “+/-” в нижнем левом углу экрана.

Примерный алгоритм работы выглядит так 

Текущая и предыдущая позиции мыши проецируются на сферу в центре сцены, заданного радиуса (радиус в дан-ном случае влияет на чувствительность мыши) в текущем состоянии (т.е. полученный ранее кватернион вращения камеры)

Считается векторное произведение векторов с началом в центре сферы и концом в спроецированных точках, и угол между этими векторами

Полученный вектор и угол используются в качестве оси вращения и угла вращения. На основе их создается ква-тернион вращения.

Текущий кватернион умножается на кватернион вращения и используется для получения матрицы вращения.

Полученная матрица используется как преобразование камеры.

Отдаление и приближение камеры происходит с помощью изменения Z координаты в части матрицы камеры, отве-чающей за ее перенос.

Тип камеры «arcball» был выбран, поскольку позволяет удобно, как с помощью мыши, так и с помощью пальца, рас-сматривать обстановку с кубами с различных сторон и степенью детальности. Кроме того, этот метод рекомендован для перспективных систем в обзоре 2016 года [4].

Работа выполнена в отделе учебной информатики ФГУ ФНЦ НИИСИ РАН по теме (проекту) "Разработка и сопровождение программного обеспечения для подготовки и проведения государственных испытаний (ЕГЭ и ОГЭ) и олимпиад по информатике и другим предметам естественнонаучного цикла для школьников и дошкольников" (0065-2015-0105).

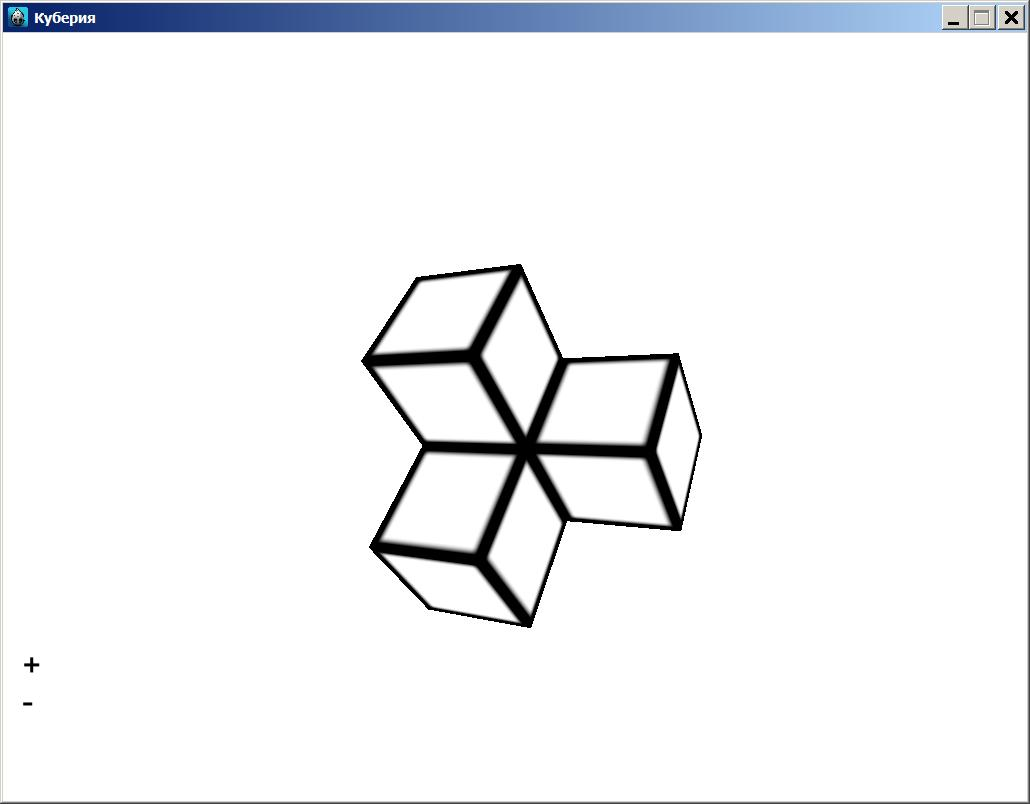
**Литература**

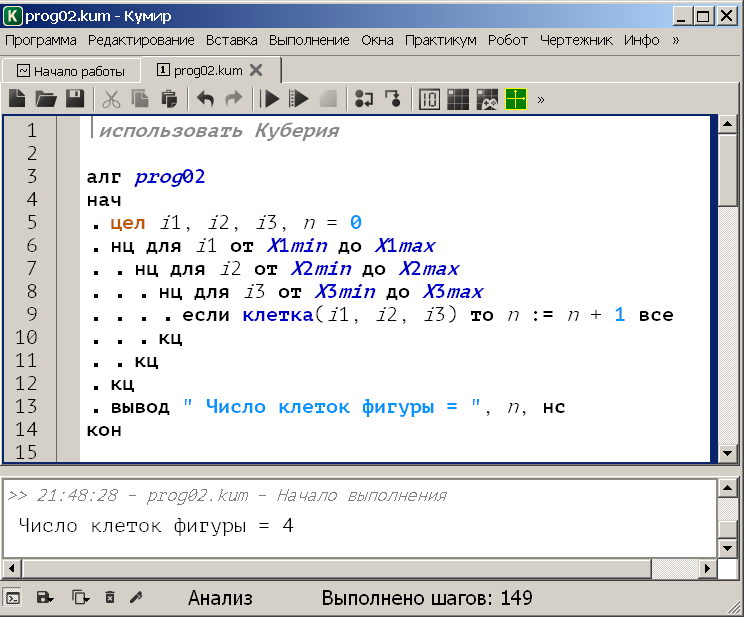
1. Ройтберг М.А., Лист в клеточку – исследовательский мир для младшеклассников, настоящий сборник. – C. 1-4.

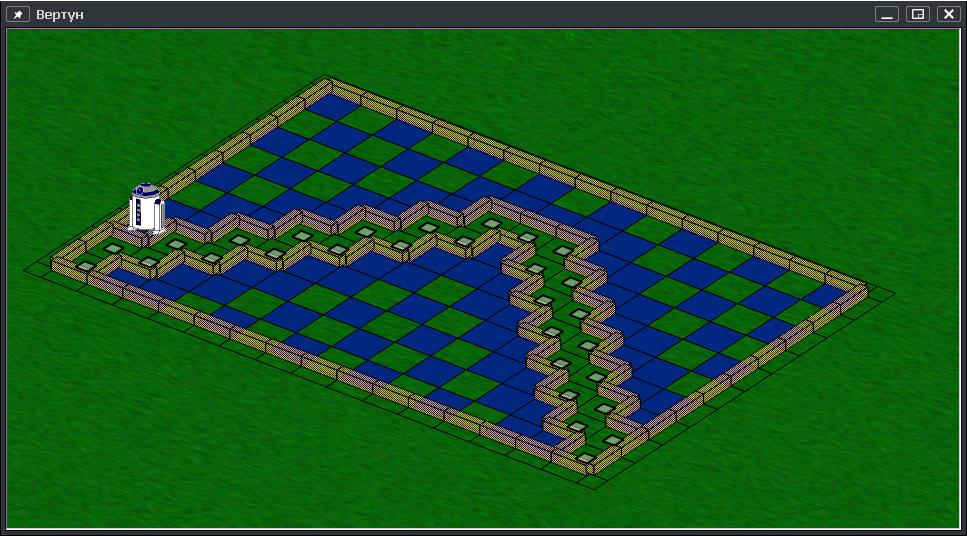
2. [http://www.cocos2d-x.org](http://www.cocos2d-x.org/)

3. A Wavefront OBJ Writer for VTK, [Doria D.,](http://www.vtkjournal.org/search/?search=authors:Doria) [The VTK Journal](http://www.vtkjournal.org/?journal=35) –2010 January – December Submissions: <http://hdl.handle.net/10380/3147>

4. Usability Comparison of Mouse, Touch and Tangible Inputs for 3D Data Manipulation Lonni Besanc¸ Paul Issartel, Mehdi Ammi, Tobias Isenberg, April 2016, <https://arxiv.org/pdf/1603.08735v2.pdf>

**Приложение** **1.** **Решение** **задачи** **3** **Первого** **цикла**



**Приложение** **2.** **Решение** **задачи** **8** **Второго** **цикла задач**

