

Глава 2. Алгоритмизация и программирование.

§ 2.3. Конструирование алгоритмов

(на примере использования Исполнителя Робот).

Описание исполнителя Робот (общие положения).

Исполнитель Робот умеет перемещаться по лабиринту, начерченному на плоскости, разбитой на клетки. Между соседними (по сторонам) клетками может стоять стена, через которую Робот пройти не может. У Робота есть девять команд. Четыре команды - это команды-приказы:

вверх вниз влево вправо

При выполнении любой из этих команд Робот перемещается на одну клетку соответственно: вверх ↑. вниз ↓. влево ←, вправо →. Если Робот получит команду передвижения сквозь стену, то он разрушится.

Также у Робота есть команда **закрасить**, при которой закрашивается клетка в которой Робот находится в настоящий момент.

Ещё четыре команды - это команды проверки условий. Эти команды проверяют, свободен ли путь для Робота в каждом из четырёх возможных направлений:

сверху свободно снизу свободно слева свободно справа свободно

Эти команды можно использовать вместе с условием «если», имеющим следующий вид:

если условие то
последовательность команд

все

Здесь *условие* – одна из команд проверки условия.

Последовательность команд - это одна или несколько любых команд-приказов.

Например, для передвижения на одну клетку вправо, если справа нет стенки, и закрашивания клетки можно использовать такой алгоритм:

если справа свободно то
вправо
закрасить

все

В одном условии можно использовать несколько команд проверки условий, применяя логические связки **и**, **или**, **не**, например:

если (справа свободно) и (не снизу свободно) то
вправо

все

Для повторения последовательности команд можно использовать цикл «пока», имеющий следующий вид:

нц пока условие
последовательность команд
кц

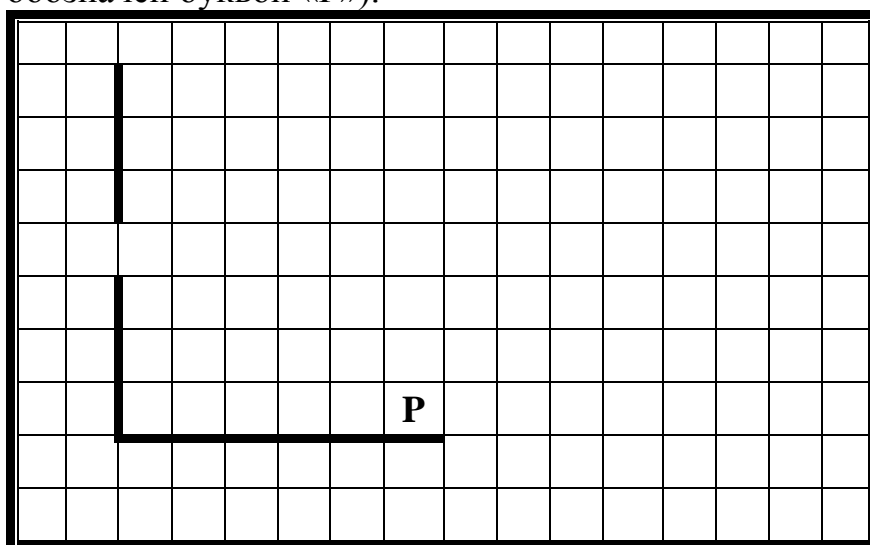
Например, для движения вправо, пока это возможно, можно использовать следующий алгоритм:

нц пока справа свободно
вправо
кц

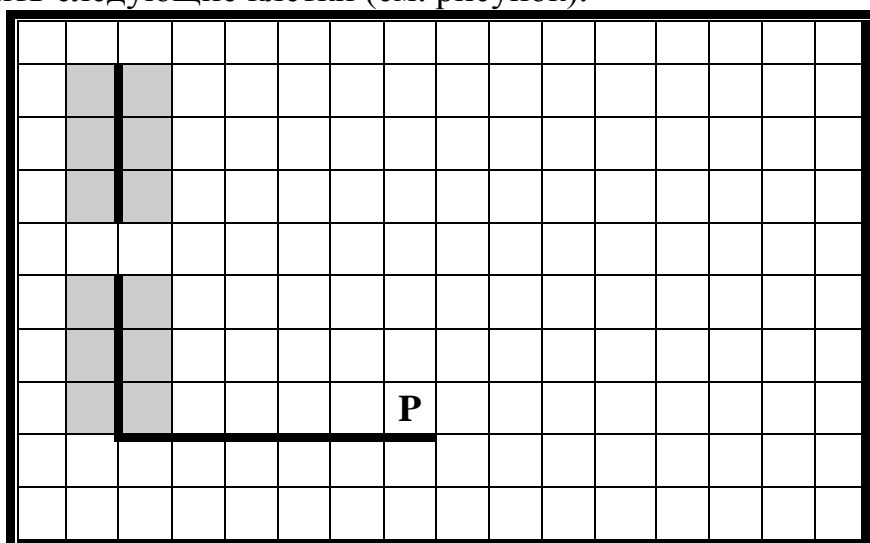
Задание.

На бесконечном поле есть горизонтальная и вертикальная стены. Левый конец горизонтальной стены соединён с нижним концом вертикальной стены. Длины стен неизвестны. В вертикальной стене есть ровно один проход, точное место прохода и его ширина неизвестны. Робот находится в клетке, расположенной непосредственно над горизонтальной стеной у её правого конца.

На рисунке указан один из возможных способов расположения стен и Робота (Робот обозначен буквой «Р»).



Напишите для Робота алгоритм, закрашивающий все клетки, расположенные непосредственно левее и правее вертикальной стены. Проход должен остаться незакрашенным. Робот должен закрасить только клетки, удовлетворяющие данному условию. Например, для приведённого выше рисунка Робот должен закрасить следующие клетки (см. рисунок).



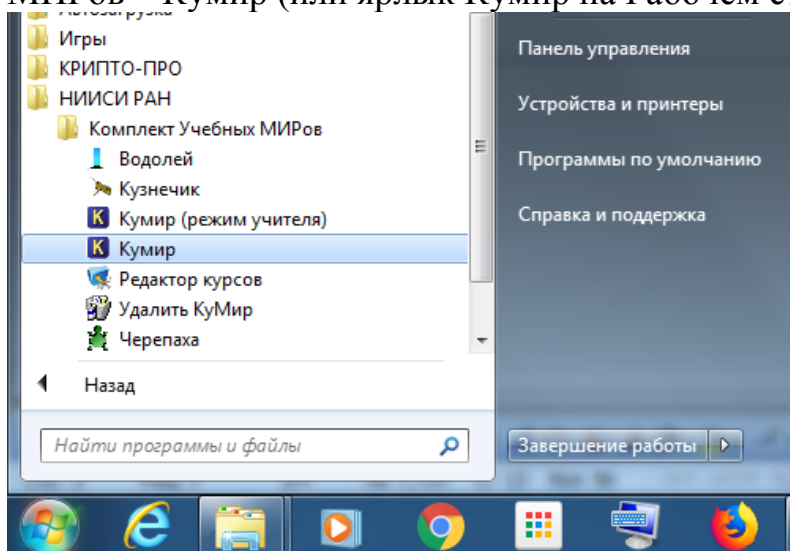
При исполнении алгоритма Робот не должен разрушиться, выполнение алгоритма должно завершиться. Конечное расположение Робота может быть произвольным. Алгоритм должен решать задачу для любого допустимого расположения стен и любого расположения и размера прохода внутри стены. Выполним алгоритм в среде формального исполнителя (для реализации мы будем использовать среду Кумир версии 1.9, возможно использование среды версии 2.0).

Алгоритм сохраняется в виде файла (с расширением *.kum*).

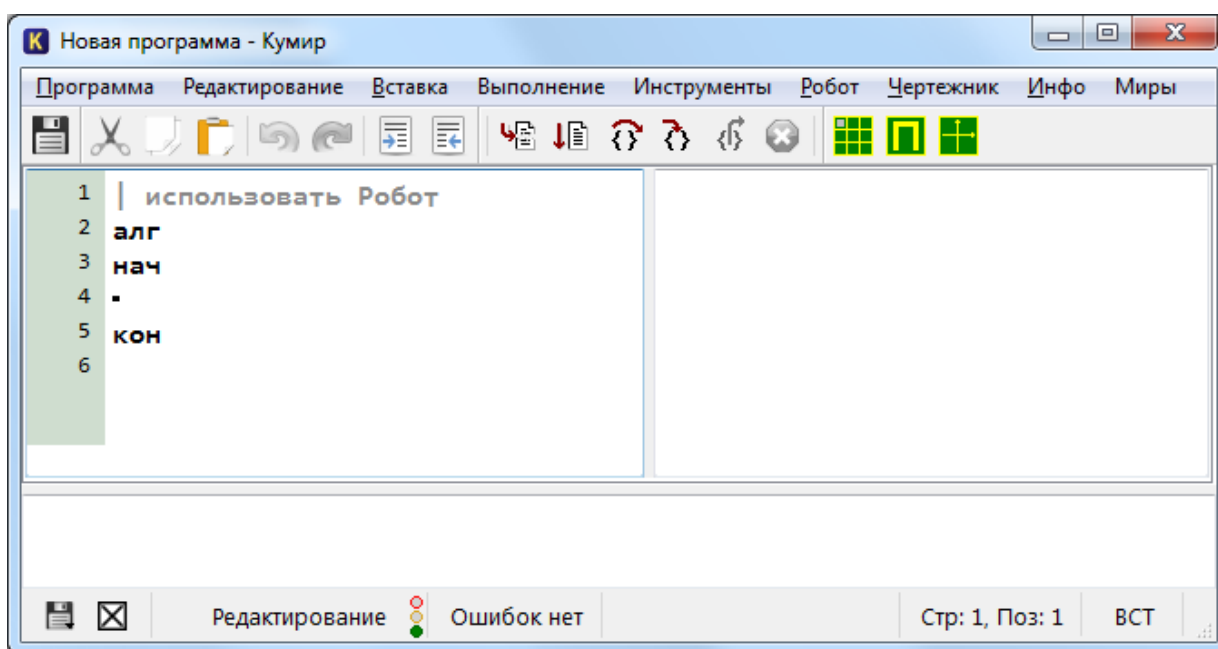
(Установочные файлы среды Кумир можно скачать по ссылке:

<https://www.niisi.ru/kumir/dl.htm>)

Запуск среды Кумир: Все программы – НИИСИ РАН – Комплект Учебных МИРов – Кумир (или ярлык Кумир на Рабочем столе):

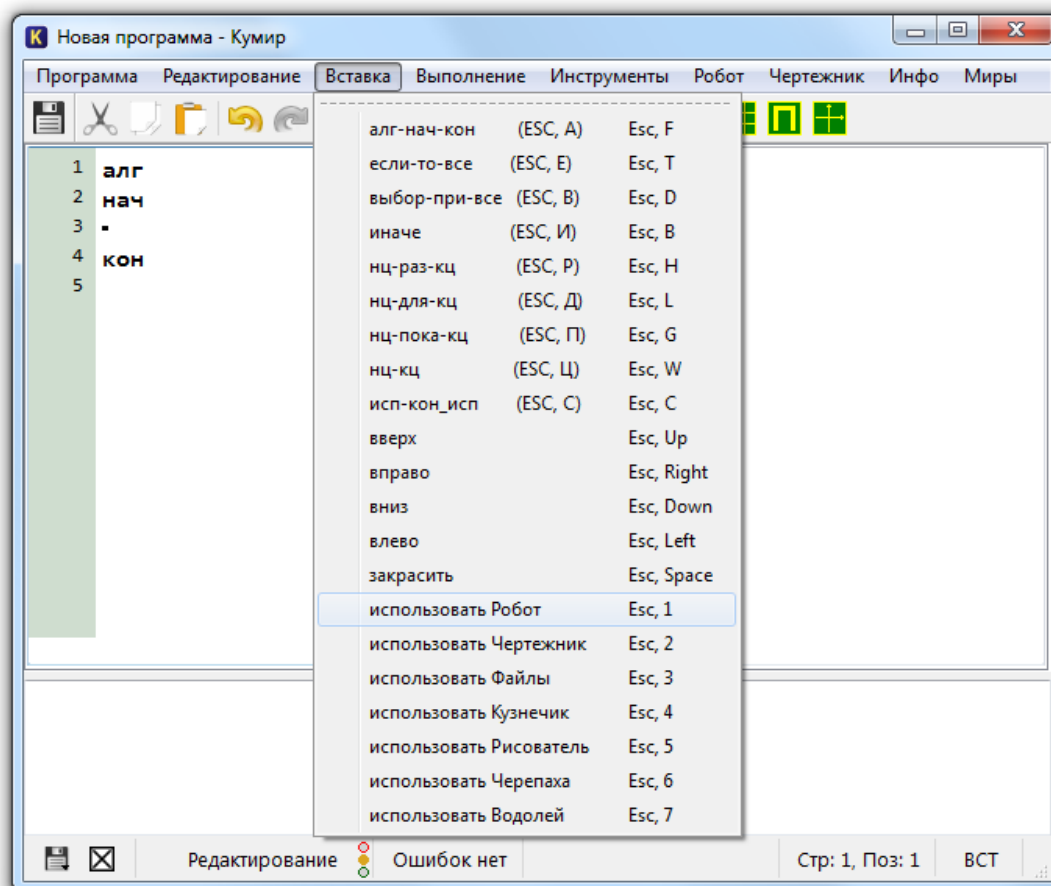


Откроется окно редактора программ (алгоритмов) среды Кумир:

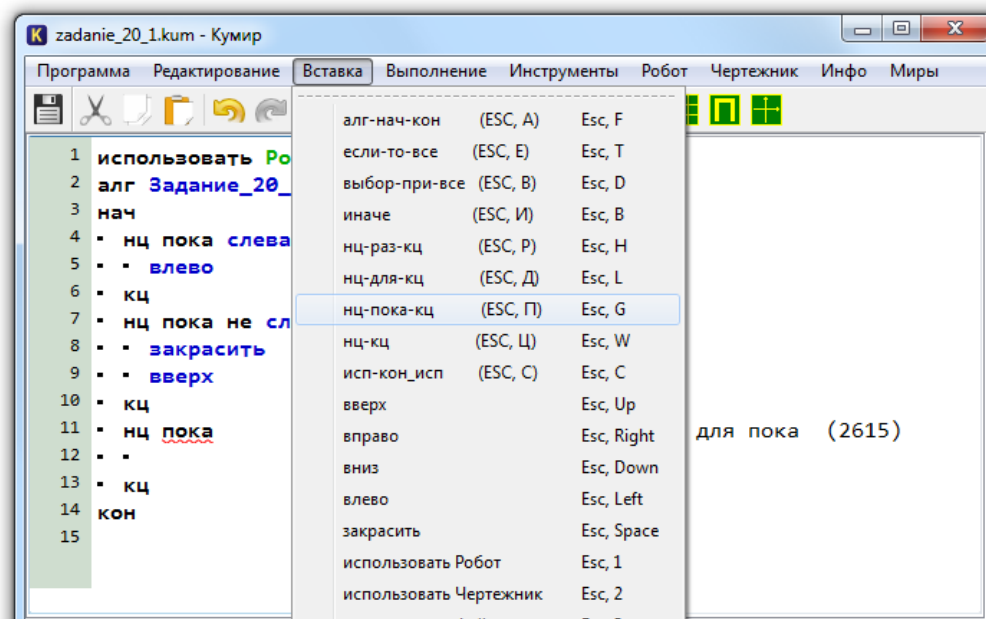


Для использования Исполнителя Робот соответствующую команду «**использовать Робот**» нужно раскомментировать – удалить символ «|» в начале строки.

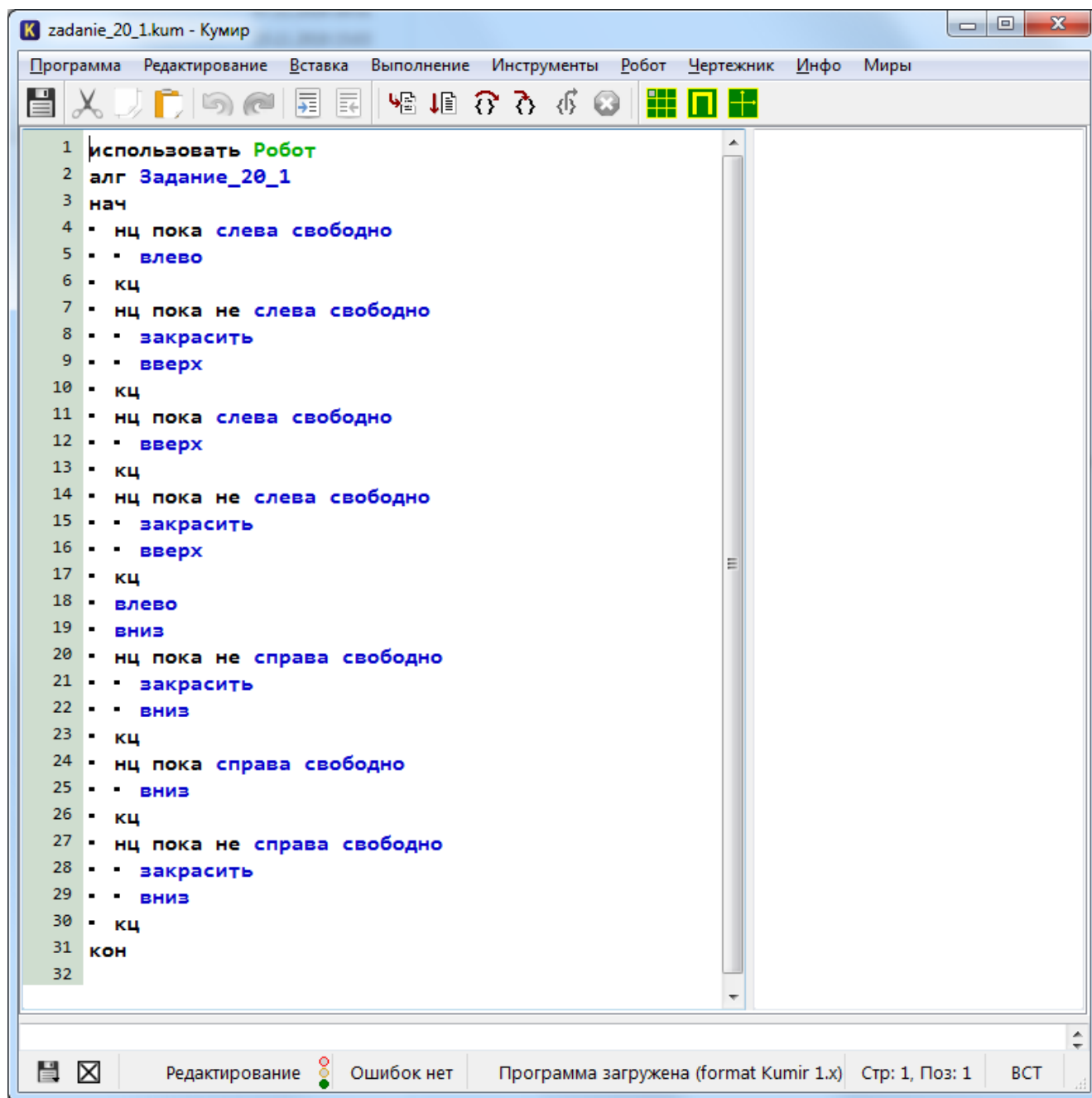
Так же, можно вставить команду «**использовать Робот**» через меню программы: *Вставка – Использовать Робот*:



Все последующие конструкции языка (циклы, команды управления Роботом) так же можно вставлять через меню *Вставка*:



Текст отредактированной программы:



Комментарии к программе:

(краткое описание последовательности действий Робота, при выполнении команд, циклов и т.п.):

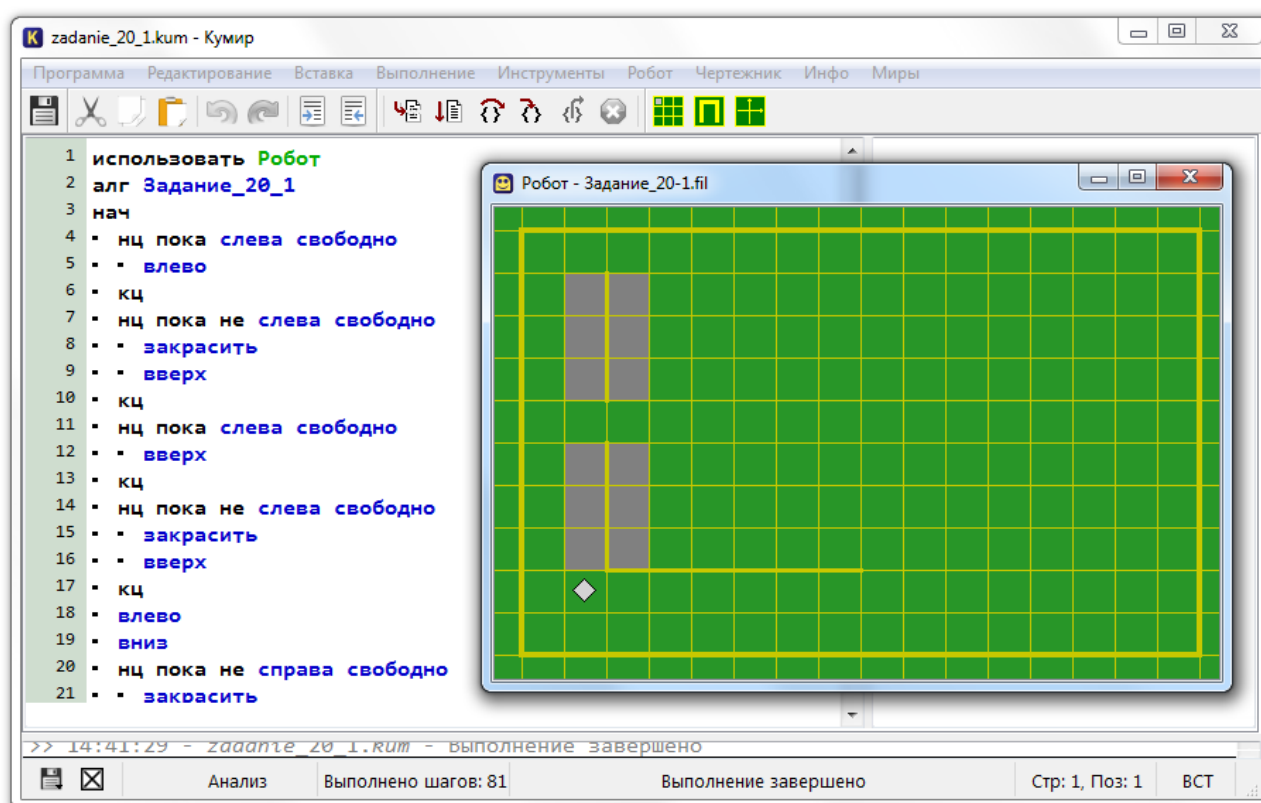
1. Робот находится в исходной точке.
2. Двигаемся влево, пока не дойдём до вертикальной стены (1-ый цикл «пока»).
3. Двигаемся вверх, пока не дойдём до прохода в стене, и закрашиваем клетки (2-ый цикл «пока»).
4. Двигаемся дальше до вертикальной стены (3-ый цикл «пока»).

5. Двигаемся вверх до конца стены и закрашиваем клетки (4-ый цикл «пока»).
6. Обходим стену (влево вниз)
7. Двигаемся вниз, пока не дойдём до прохода в стене, и закрашиваем клетки (5-ый цикл «пока»).
8. Двигаемся дальше до вертикальной стены (6-ый цикл «пока»).
9. Двигаемся вниз до конца стены и закрашиваем клетки (7-ый цикл «пока»).
10. Робот останавливается (программа завершена).

Замечания:

- в программе используется цикл «пока», т.к. он позволяет производить действия Робота по условию: как правило, это наличие или отсутствие свободного пространства в нужном направлении;
- мы не знаем длину стен, т.о. использовать циклы «для» или «N раз» нельзя;
- в программе не используются вспомогательные алгоритмы (подпрограммы), хотя такой вариант возможен (есть по два повторяющихся цикла).

Запуск и выполнение программы:

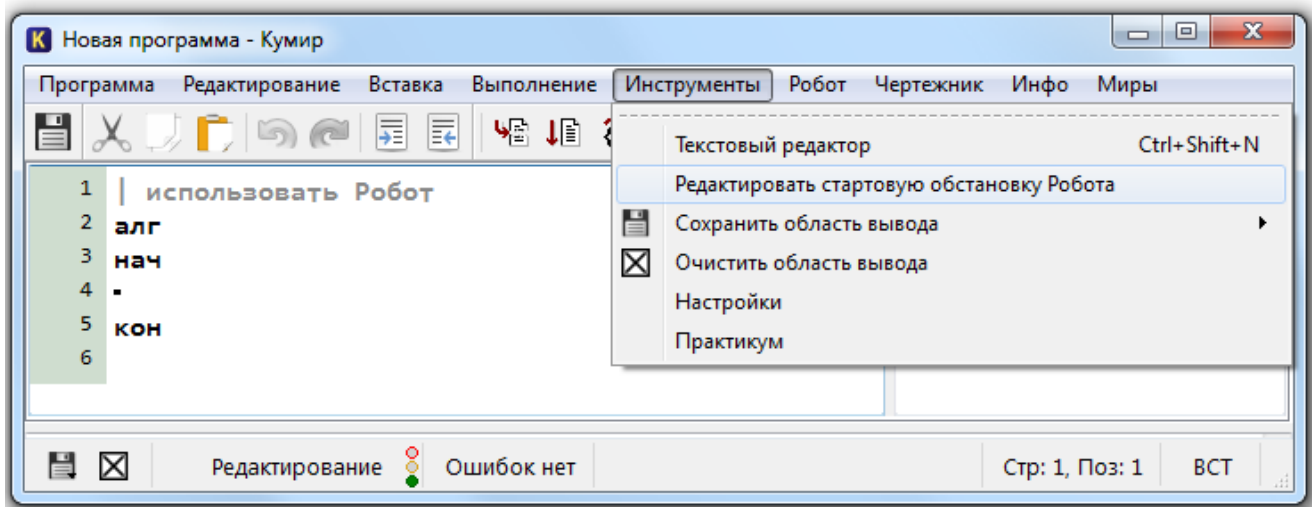


Замечание:

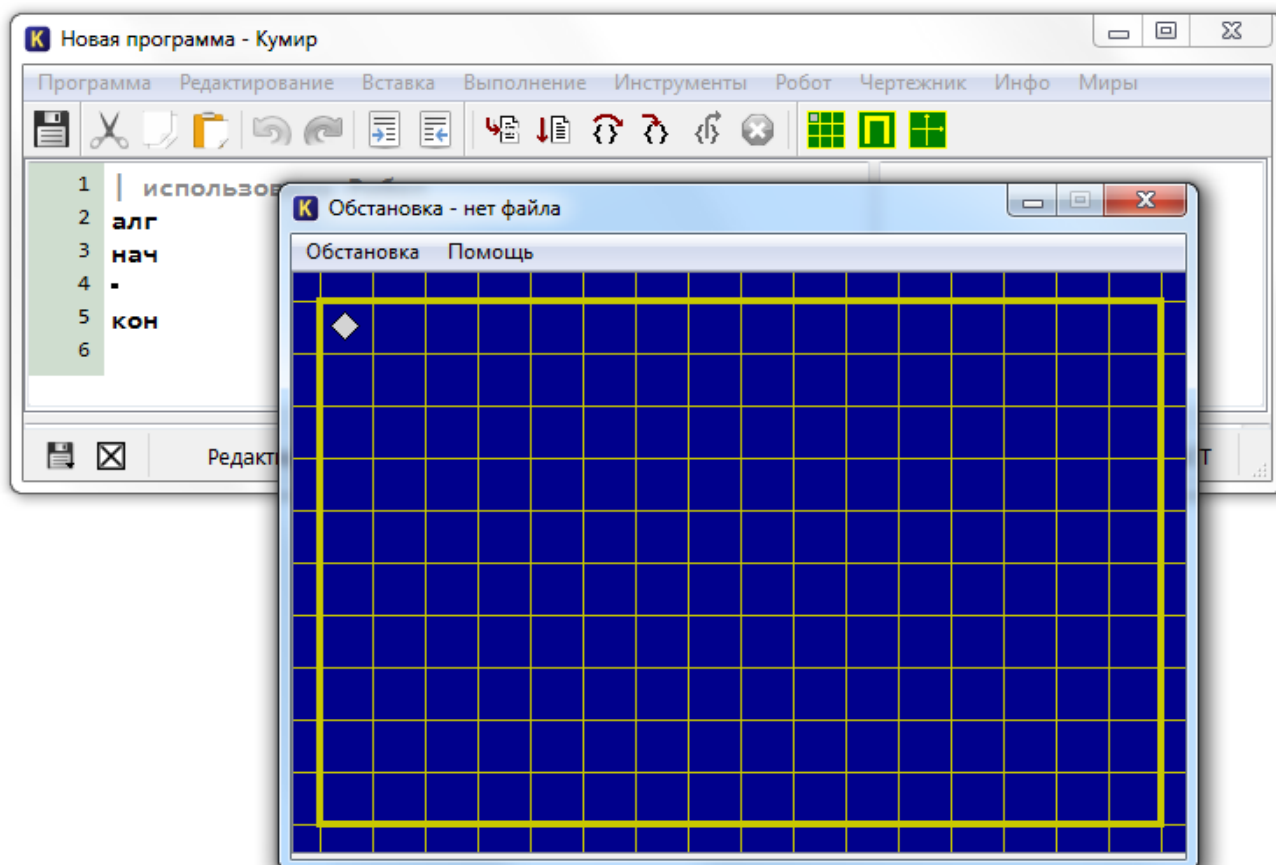
Перед запуском программы необходимо отредактировать и сменить стартовую обстановку Робота (если это не было сделано ранее).

Редактирование стартовой обстановки Робота:

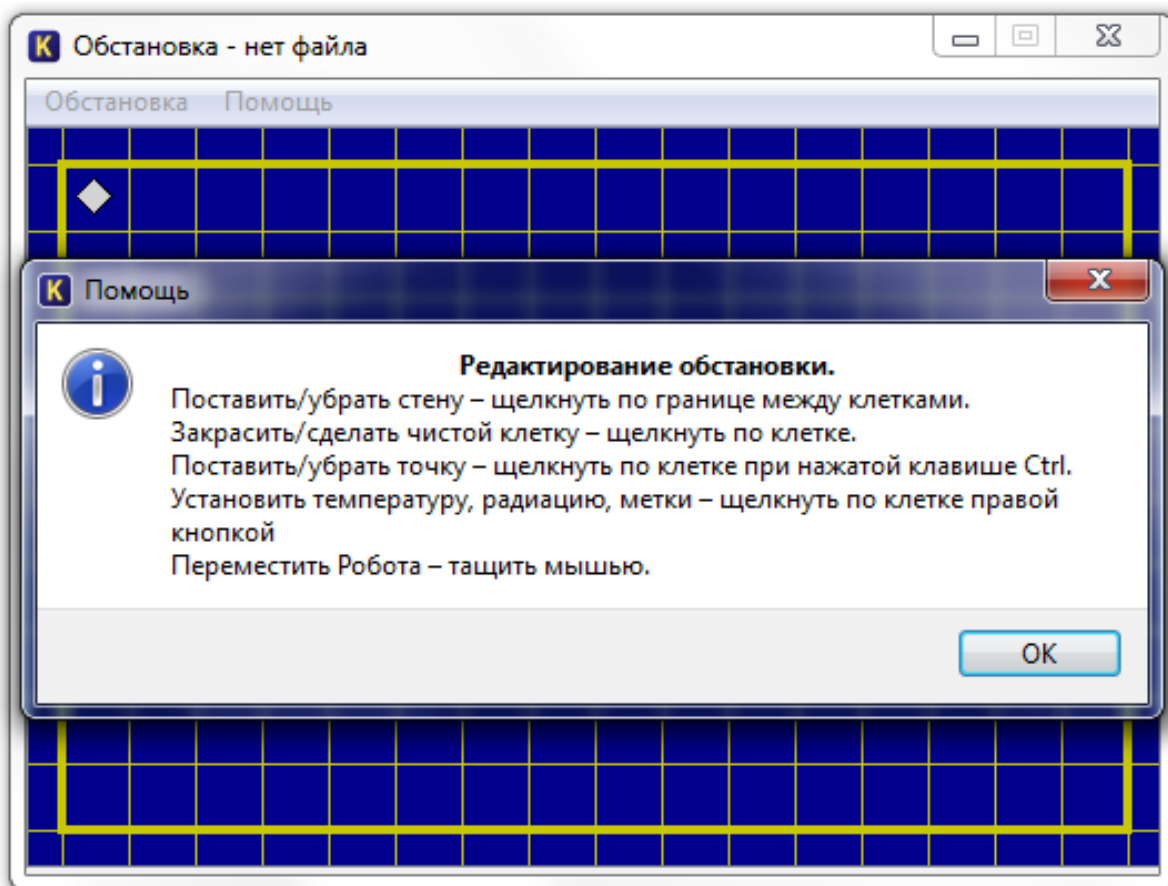
Вызывается командой из меню: *Инструменты* – *Редактировать стартовую обстановку Робота*:



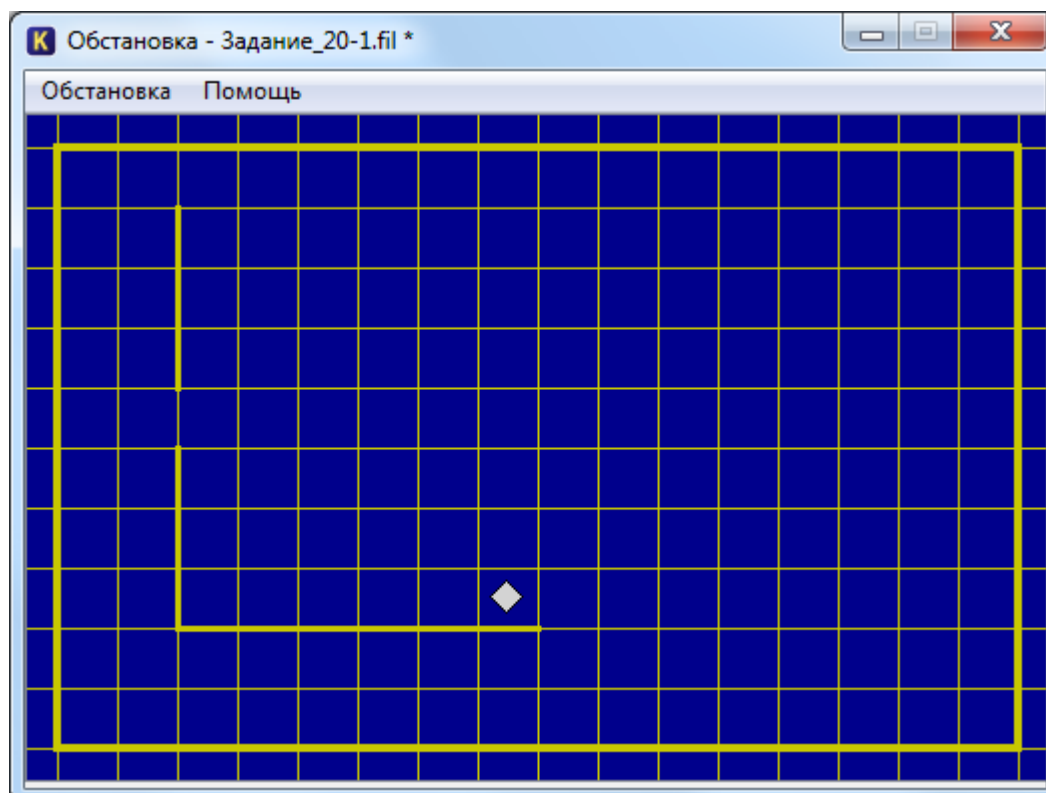
Откроется редактор обстановки (поле Робота будет синего цвета; при выполнении программ – оно зеленое):



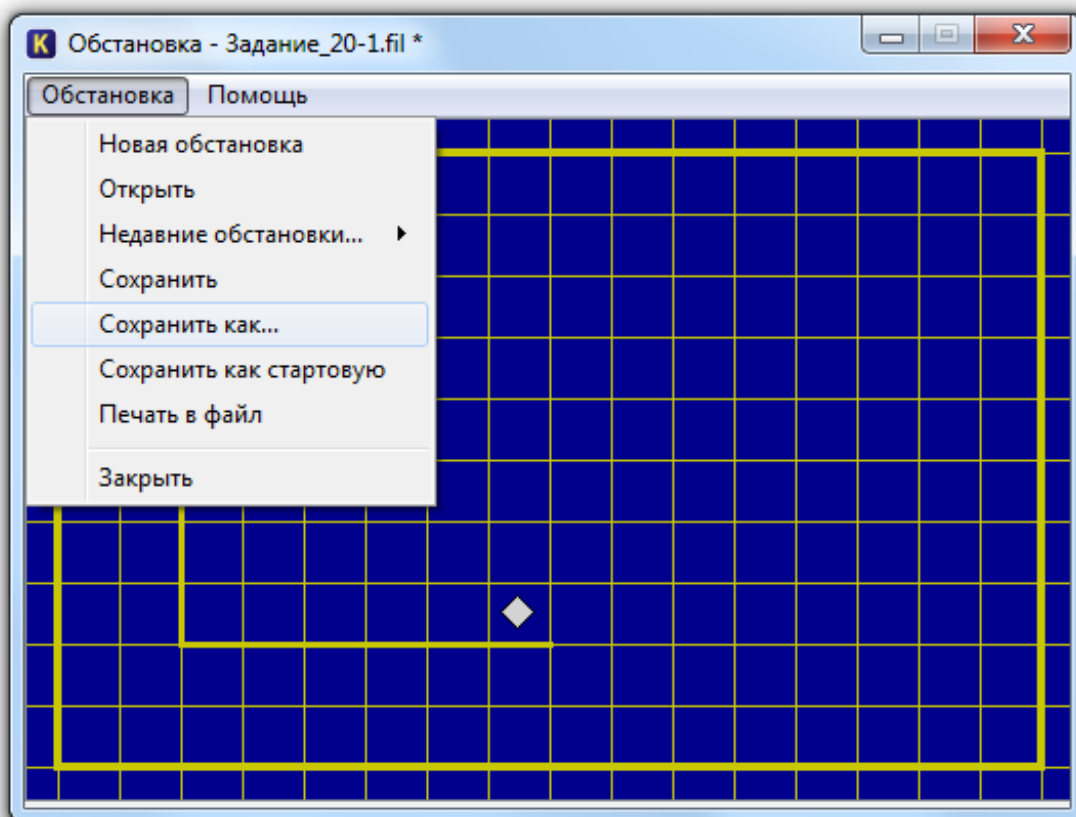
Если нужна информация по командам редактирования – нужно выбрать пункт *Помощь* в меню:



Т.о. – все команды редактирования осуществляются щелчком и перетаскиванием мышью (Робот обозначен как - ◆) .
 В нашем примере, отредактированная обстановка Робота будет выглядеть так:

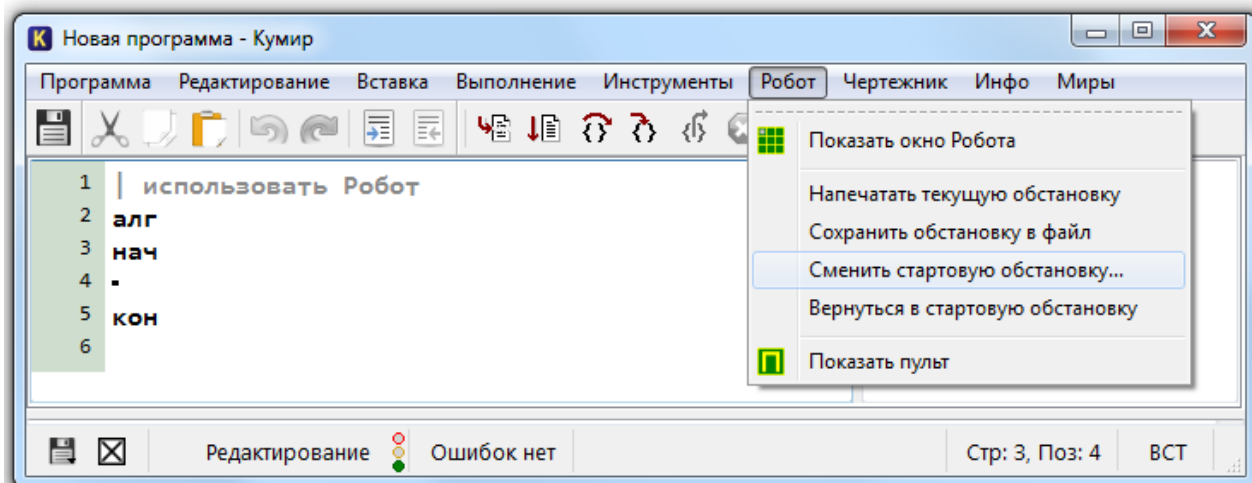


Отредактированную обстановку нужно сохранить в виде файла. Это делается командой: *Обстановка – Сохранить как:*



По умолчанию, система присвоит файлу расширение **.fil** (изменять его не следует). Имя может быть произвольным. В нашем примере – это *Задание_20-1.fil*.

И самое главное: перед выполнением программы нужно загрузить отредактированную обстановку из файла. Это делается уже из меню самой программы Кумир: *Робот – Сменить стартовую обстановку* (и указать путь к файлу с расширением **.fil** , в котором вы сохранили эту самую обстановку).



Литература:

- 1) **Информатика** : учебник для 9 класса / Босова Л.Л., Босова А.Ю. — М. : БИНОМ. Лаборатория знаний, 2013. — 184 с. : ил. - § 2.3, вопросы и задания (стр. 76-88).
- 2) Босова Л. Л. **Информатика : рабочая тетрадь** для 9 класса : в 2 ч. Ч 2. / Л. Л. Босова, А. Ю. Босова. — М. : БИНОМ. Лаборатория знаний, 2016. — 96 с.: ил. – Задания №№ 84-86 (стр. 6-12).
- 3) **Информатика** : учебник для 8 класса / Босова Л.Л., Босова А.Ю. — М. : БИНОМ. Лаборатория знаний, 2013. — 160 с. : ил. - § 2.1-2.4, вопросы и задания (стр. 46-96).
- 4) При подготовке использовался: Демонстрационный вариант контрольных измерительных материалов для проведения в 2019 году основного государственного экзамена по ИНФОРМАТИКЕ и ИКТ (подготовлен Федеральным государственным бюджетным научным учреждением «ФЕДЕРАЛЬНЫЙ ИНСТИТУТ ПЕДАГОГИЧЕСКИХ ИЗМЕРЕНИЙ»), задание по выбору - 20.1/ © Федеральная служба по надзору в сфере образования и науки Российской Федерации, 2019.

Программное обеспечение:

Среда программирования Кумир (версия 1.9.x)

Задание на практическую работу:

- 1) Напишите для Робота алгоритм, закрашивающий все клетки, расположенные относительно стен (размер которых неизвестен) разделенных проходом(и) который должен остаться незакрашенным. Робот должен закрасить только клетки, удовлетворяющие условию задачи.
- 2) Для выполнения задачи, отредактируйте и сохраните «обстановку Робота» в файл.