

Содержание

Методические указания к выполнению организуемой работы студента	3
Порядок выполнения работы	3
Требования к предоставляемым материалам	5
Библиотека вспомогательных алгоритмов, рекомендуемых к использованию	7
Закрашенные области	9
Вариант 1	9
Вариант 2	10
Вариант 3	11
Вариант 4	12
Вариант 5	13
Вариант 6	14
Вариант 7	15
Вариант 8	16
Вариант 9	17
Вариант 10	18
Вариант 11	19
Вариант 12	20
Вариант 13	21
Вариант 14	22
Задачи поиска источников радиации	23
Уровень 1	23
Вариант 1	23
Вариант 2	24
Вариант 3	25
Вариант 4	26
Вариант 5	27
Уровень 2	29
Вариант 6	29
Вариант 7	30
Вариант 8	31
Вариант 9	32
Вариант 10	33
Варианты заданий	36
Уровень 1	36

Варианты 1 - 19	36
Варианты 20 - 39	37
Варианты 40 - 56	38
Уровень 2	39
Варианты 77 - 96	40
Варианты 97 - 116	41
Варианты 117 - 140	42

Методические указания к выполнению организуемой работы студента

Тема «Поиск роботом источников радиации в закрашенной области»


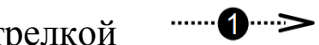
Цель работы — усовершенствовать навыки:

- анализа и исследования задачи;
- построения алгоритмов с использованием типовых алгоритмических структур;
- использования вспомогательных алгоритмов для устранения избыточности кода.

Порядок выполнения работы

Примечание: звездочкой (*) помечены обязательные требования, которые нельзя игнорировать. Их невыполнение приводит к неудовлетворительной оценке за работу.

1. *Ознакомиться с заданием согласно варианту, выданному преподавателем. Задание предполагает поиск и анализ источников радиации в заданной закрашенной области. Вариант задания формируется путем комбинации задачи поиска источников радиации и вида закрашенной области. Закрашенная область изначально присутствует в обстановке.
2. Выполнить анализ и исследование поставленной задачи, как результат сформировать таблицу ситуаций и идей их решения. Рекомендуемая последовательность действий:
 - а) *Выделить независимые подзадачи, которые могут решаться последовательно. Для каждой подзадачи выполнить следующие действия.
 - б) *Создать стартовые обстановки (ситуации), отличающиеся по решению задачи/подзадачи, размеру/форме поля (закрашенной области), количеству и/или расположению источников радиации. Для каждой обстановки (ситуации) дать наименование, отличающее ее от других и отражающее ее суть.
 - в) Показать решение задачи/подзадачи в виде стрелок, которые

соответствуют перемещению робота в непрерывном ряду. Стрелкой  отобразить перемещение робота без поиска источников радиации, а стрелкой  перемещение робота в ряду с поиском источников радиации. Номер выделяет конкретное действие среди других; для разных ситуаций одно и то же действие должно иметь одинаковый номер. Также номер определяет порядок выполнения действий.

- d) *В идее решения указать какие действия требуется выполнить для достижения результата и почему.
3. *Построить алгоритм решения задачи, обобщив решения для разных ситуаций. При построении алгоритма использовать только типовые алгоритмические структуры (смотри признаки применения и порядок декомпозиции алгоритмических структур). В алгоритме **ЗАПРЕЩЕНО** использовать команды прерывающие естественный ход программы, например, команды «выход», «стоп» и т.д.

Алгоритм представить в графической нотации, используя диаграммы активности среды Astah. При изображении типовых алгоритмических структур использовать только их канонические формы — объединение на одной диаграмме нескольких алгоритмических структур не допускается. Простые действия должны объединяться в составные в соответствии со смыслом решаемой задачи.

4. *Если в алгоритме встречаются составные действия, которые могут быть решены «библиотечными» вспомогательными алгоритмами, то это указывается явно, а сами действия не декомпозируются.
5. *При составлении алгоритма необходимо определиться с вариантом «обязательного» вспомогательного алгоритма. Выбранный вариант должен обеспечивать простоту решения задачи.
6. *При необходимости, составное действие в алгоритме можно реализовать как «дополнительный» вспомогательный алгоритм, если оно удовлетворяет следующим условиям:
- a) действие встречается два и более раз, или имеются похожие по смыслу реализации;

- b) действие имеет завершённый характер, т.е. четко определены ее назначение, входные и выходные данные;
 - c) действие имеет универсальный характер, т.е. может использоваться в задачах подобного типа;
 - d) действие имеет минимальный интерфейс (не более 3 входных и не более 3 выходных данных), а сам вспомогательный алгоритм помещается на экране (не более 40 строк), но и не слишком маленький (не менее 5 строк).
7. *Построить диаграмму вызовов вспомогательных алгоритмов в виде диаграммы компонентов среды Astah.
8. *Закодировать алгоритм решения задачи в среде КуМир. Код программы должен быть эквивалентен алгоритму программы, то есть:
- порядок действий в алгоритме и программе должен совпадать;
 - каждой типовой алгоритмической структуре «ветвление» и «цикл» должна соответствовать команда ветвления/цикла;
 - элементарному действию алгоритма должна соответствовать простая команда языка КуМир;
 - составному действию алгоритма должен соответствовать комментарий.
9. *Запустить и отладить программу на ранее созданных обстановках. При необходимости скорректировать таблицу, содержащую результаты анализа и исследования задачи, алгоритм решения задачи и программный код.

Требования к предоставляемым материалам

1. *Необходимо сдать электронную и печатную версии протокола семестровой работы. Протокол должен содержать:
- a) титульный лист с указанием своей специальности, ФИО, группы, варианта и даты сдачи;
 - b) формулировку задания;
 - c) результаты исследования и анализа (таблица ситуаций и идеи решения);
 - d) диаграмму вызова вспомогательных алгоритмов;
 - e) алгоритм решения задачи;

f) программный код.

Все страницы, кроме первой должны быть пронумерованы (внизу по центру страницы). Электронная версия протокола семестровой работы должна быть в формате doc (Microsoft Word 97/2000/XP) или odt (OpenOffice 3+, LibreOffice 4+).

2. Электронная версия должна содержать:

- a) протокол семестровой работы;
- b) файл, созданный в среде Astah, содержащий диаграмму вызова вспомогательных алгоритмов, и диаграммы активности для каждого вспомогательного алгоритма и основного алгоритма;
- c) программный код, созданный в среде КуМир;
- d) набор обстановок для тестирования программы. Файлы с обстановками должны быть помещены в отдельную папку.

Все файлы и папки должны быть помещены в главную папку, названную по ФИО студента. Пример содержания папок:

- Сидоров Иван Петрович
 - Протокол семестровой работы.doc
 - Алгоритм.asta
 - Поиск радиации.kum
 - Обстановки:
 - Обстановка_1.fil
 - Обстановка_2.fil
 -
 - Обстановка_N.fil

Библиотека вспомогательных алгоритмов, рекомендуемых к
использованию

| Дойти до стены в заданном направлении

алг **цел** дойти_до_стены(арг **лит** направление)

| робот в произвольной клетке, направление перемещения

. дано направление = "**вверх**" или направление = "**вниз**" или направление = "**влево**"
или направление = "**вправо**"

. надо стена_рядом(направление) | робот рядом со стеной в указанном
направлении;

| кол-во пройденных шагов

нач

. знач := 0

. нц пока не стена_рядом(направление)

. . | Перейти на следующую клетку

. . перейти_на_клетку(направление)

. .

. . знач := знач + 1

. кц

кон

| Определить, имеется ли рядом стена в заданном направлении

алг **лог** стена_рядом(арг **лит** направление)

| робот в произвольной клетке, направление поиска стены

. дано направление = "**вверх**" или направление = "**вниз**" или направление = "**влево**"
или направление = "**вправо**"

. надо | признак наличия стены в заданном направлении

нач

. выбор

. . при направление = "**влево**": знач:= слева стена

. . при направление = "**вправо**": знач:= справа стена

. . при направление = "**вверх**": знач:= сверху стена

. . при направление = "**вниз**": знач:= снизу стена

. все

кон

| Перейти на клетку в заданном направлении.
| Считается, что стены не мешают роботу передвигаться
алг перейти_на_клетку(арг **лит** направление)
| робот в произвольной клетке; известно направление перемещения;
| предполагается, что стены не мешают роботу перемещаться
. дано направление = "**вверх**" или направление = "**вниз**" или направление = "**влево**"
или направление = "**вправо**"

. надо | робот в позиции, отстоящей от исходной на одну клетку в заданном
| направлении

нач

. выбор

. . при направление = "**влево**": влево
. . при направление = "**вправо**": вправо
. . при направление = "**вверх**": вверх
. . при направление = "**вниз**": вниз

. все

кон

| Идти в заданном направлении заданное кол-во шагов.
| Считается, что стены не мешают роботу передвигаться
алг идти_кол_во_шагов(**лит** направление, **цел** кол_во_шагов)
| робот в произвольной позиции, известно направление движения и кол-во шагов;
| стены не мешают роботу передвигаться
. дано кол_во_шагов ≥ 0 и (направление = "**вверх**" или направление = "**вниз**" или
направление = "**влево**" или направление = "**вправо**")

. надо | робот в позиции, удаленной от исходной на заданное кол-во шагов

нач

.

. нц кол_во_шагов раз

. . | Переходим на следующую клетку
. . перейти_на_клетку(направление)

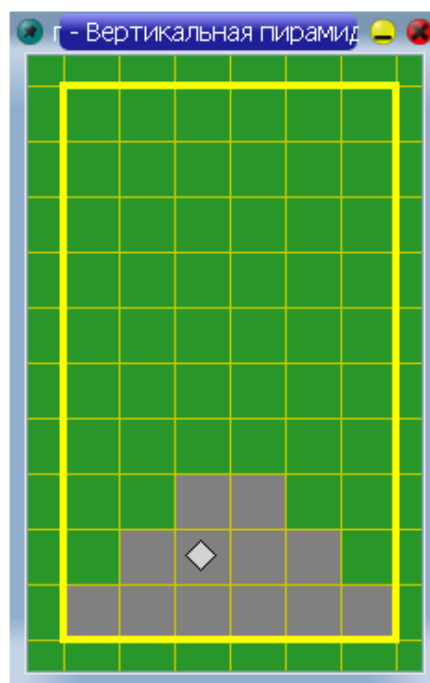
. кц

.

кон

Закрашенные области

Вариант 1



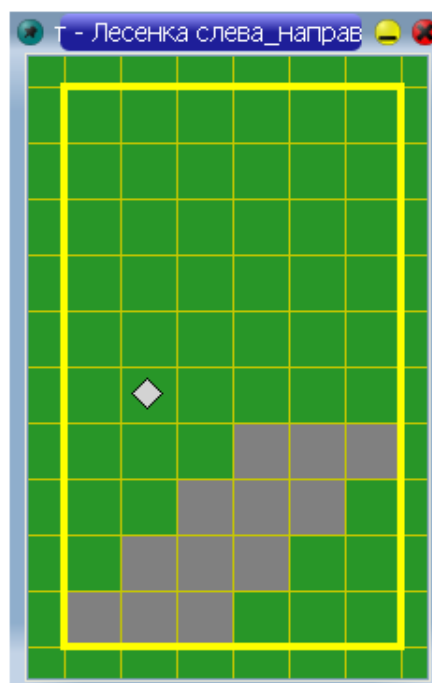
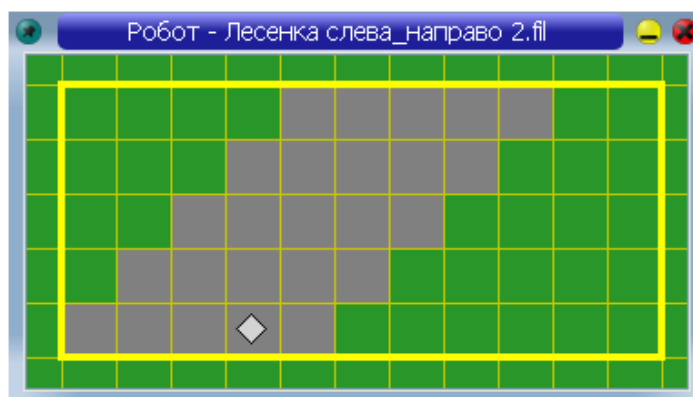
Размеры поля могут варьироваться в пределах:

$3 \leq \text{ширина} \leq 11, 3 \leq \text{высота} \leq 10$

Способ построения:

Нижняя закрашенная строка области равна ширине поля. Каждая следующая закрашенная строка на две клетки меньше чем предыдущая строка и отцентрирована по ширине поля.

Вариант 2



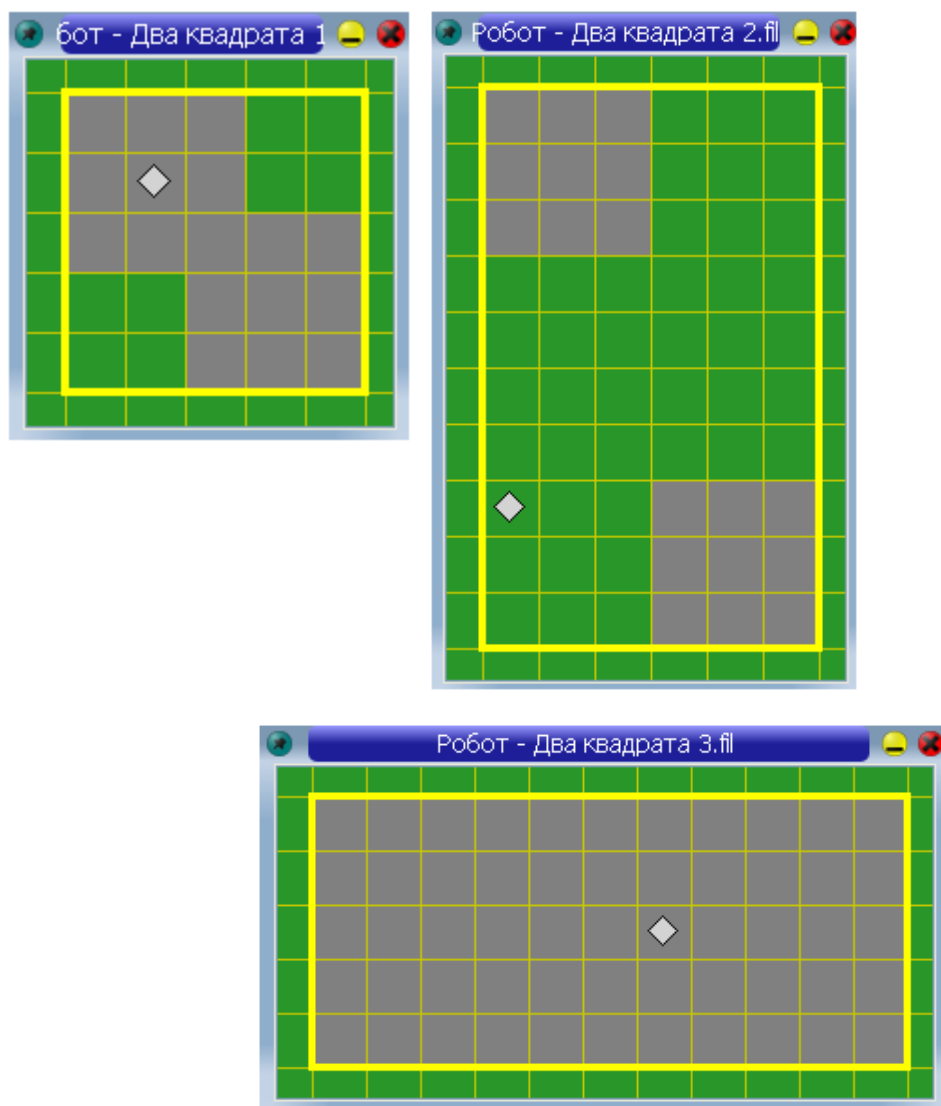
Размеры поля могут варьироваться в пределах:

$$4 \leq \text{ширина} \leq 11, 4 \leq \text{высота} \leq 10$$

Способ построения:

Нижняя закрашенная строка области равна (ширине поля)/2 (если получается дробное значение, то оно округляется вниз до целого значения) и привязана к левой границе поля. Каждая следующая закрашенная строка равна предыдущей строке, но смещена относительно нее на одну клетку вправо.

Вариант 3



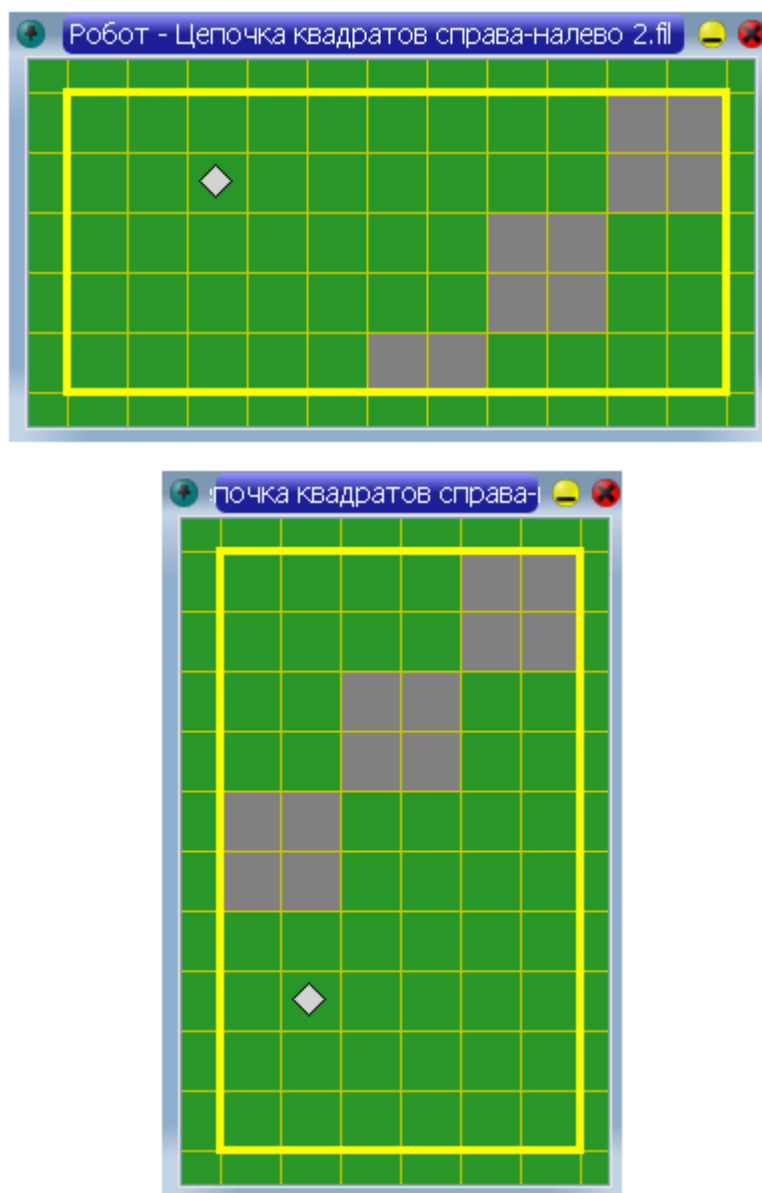
Размеры поля могут варьироваться в пределах:

$$4 \leq \text{ширина} \leq 11, 4 \leq \text{высота} \leq 10$$

Способ построения:

Закрашенная область состоит из двух квадратов одинакового размера. Размер квадрата равен $(\text{ширине поля})/2$ (если получается дробное значение, то оно округляется вверх до целого значения). Один квадрат привязан к левому верхнему углу поля, а другой — к правому нижнему.

Вариант 4



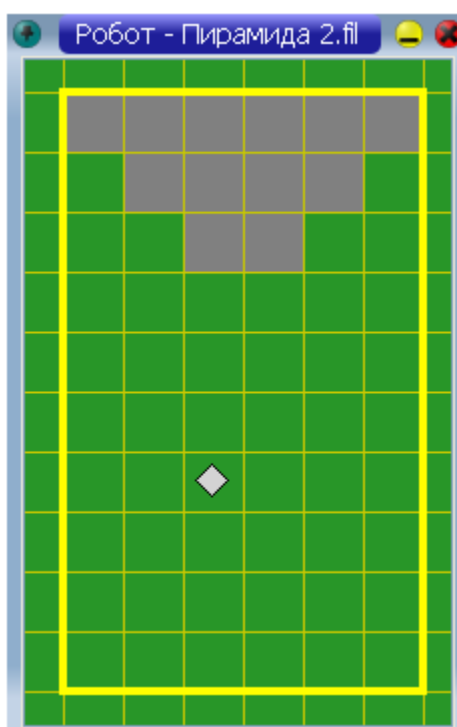
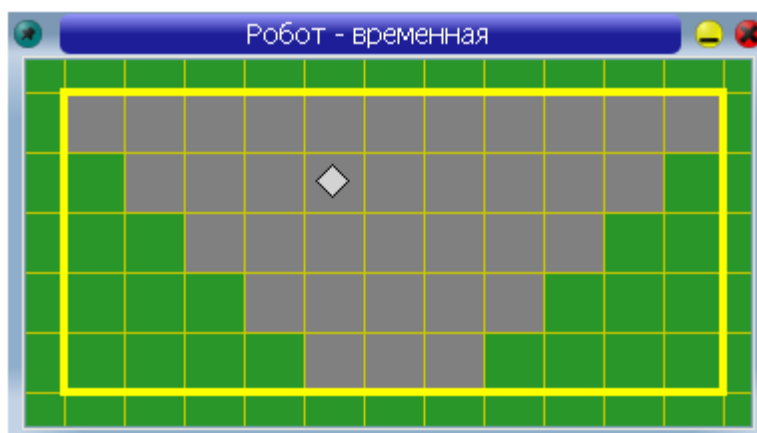
Размеры поля могут варьироваться в пределах:

$3 \leq \text{ширина} \leq 11, 3 \leq \text{высота} \leq 10$

Способ построения:

Закрашенная область состоит из цепочки квадратов размера 2x2.
Цепочка квадратов начинается с правого верхнего угла поля.

Вариант 5



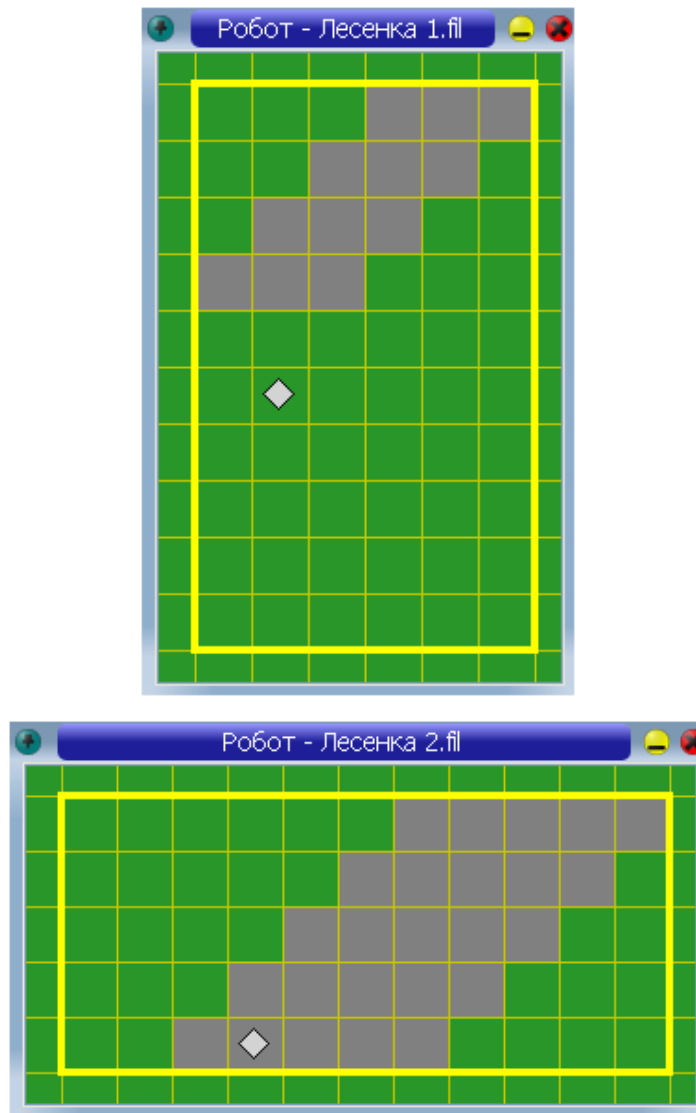
Размеры поля могут варьироваться в пределах:

$3 \leq \text{ширина} \leq 11, 3 \leq \text{высота} \leq 10$

Способ построения:

Верхняя закрашенная строка области равна ширине поля. Каждая следующая закрашенная строка на две клетки меньше чем предыдущая строка и отцентрирована по ширине поля.

Вариант 6



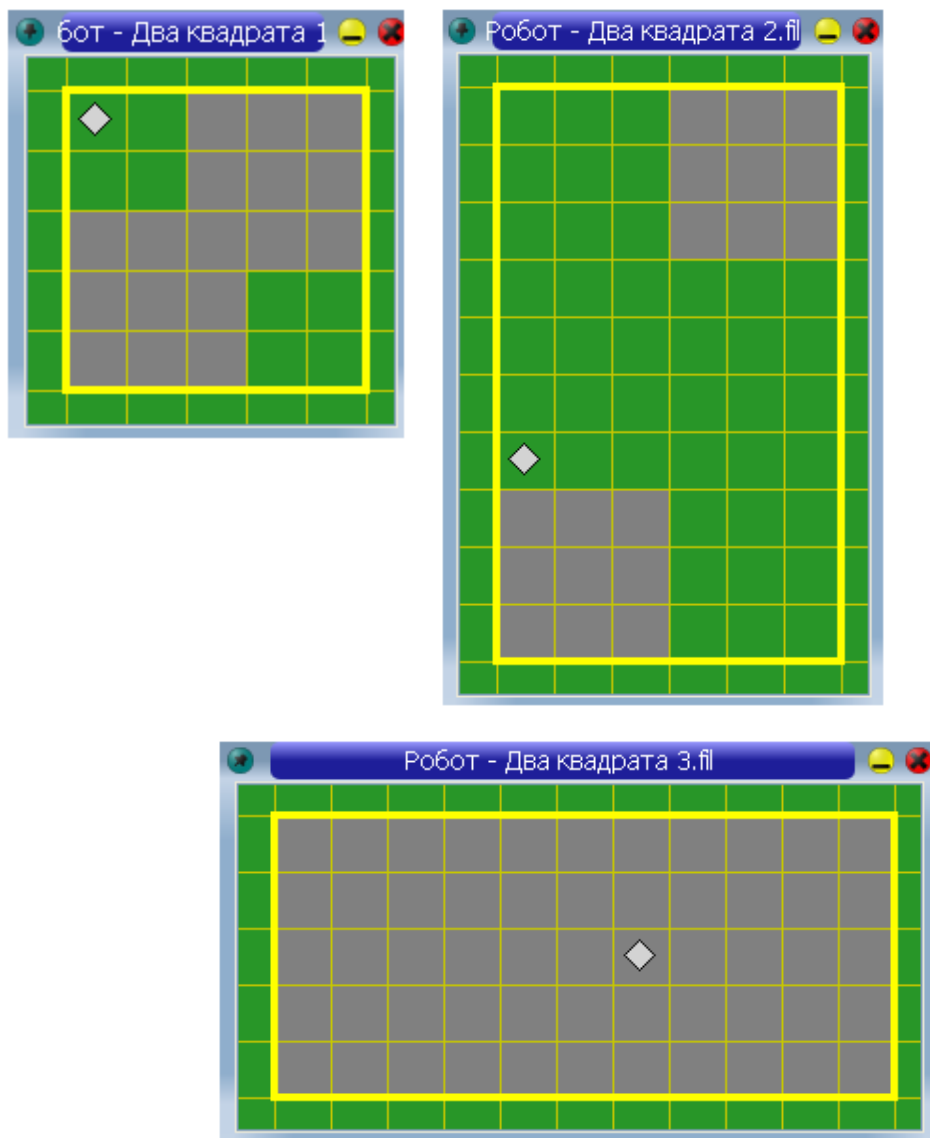
Размеры поля могут варьироваться в пределах:

$$4 \leq \text{ширина} \leq 11, 4 \leq \text{высота} \leq 10$$

Способ построения:

Верхняя закрашенная строка области равна (ширине поля)/2 (если получается дробное значение, то оно округляется вниз до целого значения) и привязана к правой границе поля. Каждая следующая закрашенная строка равна предыдущей строке, но смещена относительно нее на одну клетку влево.

Вариант 7



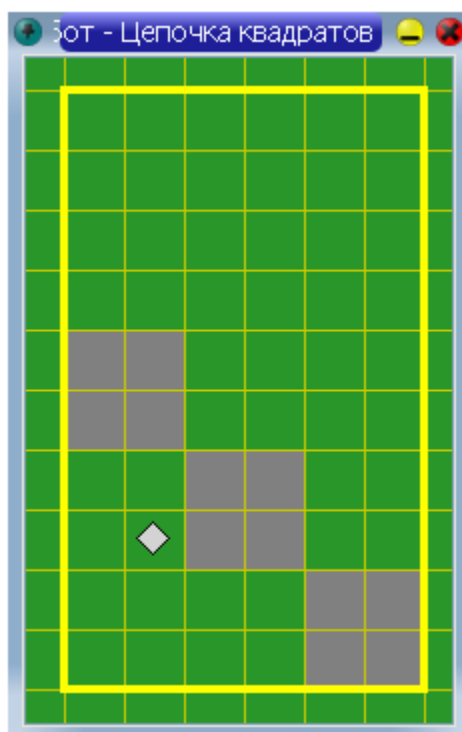
Размеры поля могут варьироваться в пределах:

$$4 \leq \text{ширина} \leq 11, 4 \leq \text{высота} \leq 10$$

Способ построения:

Закрашенная область состоит из двух квадратов одинакового размера. Размер квадрата равен $(\text{ширине поля})/2$ (если получается дробное значение, то оно округляется вверх до целого значения). Один квадрат привязан к левому нижнему углу поля, а другой — к правому верхнему.

Вариант 8



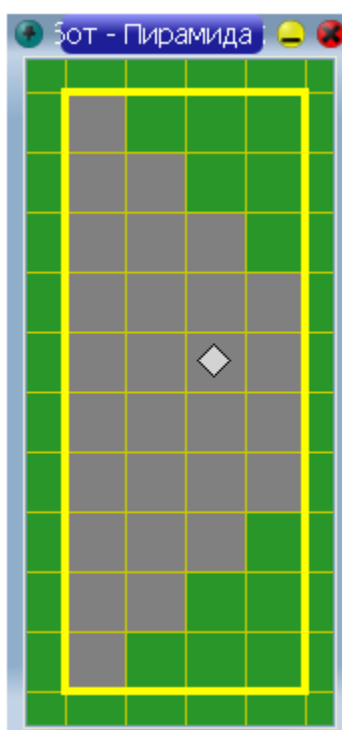
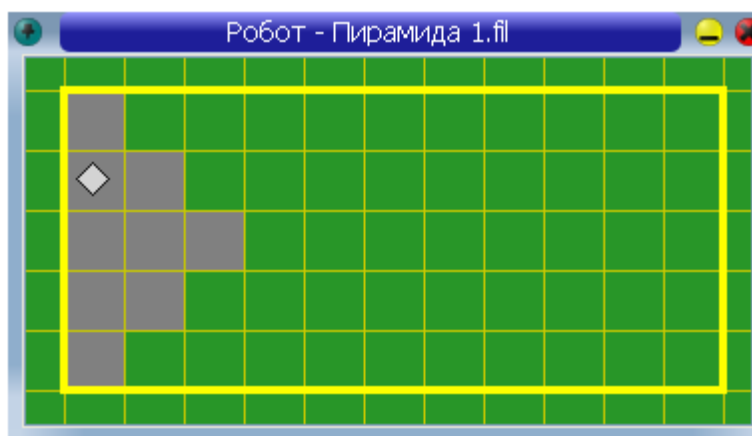
Размеры поля могут варьироваться в пределах:

$3 \leq \text{ширина} \leq 11$, $3 \leq \text{высота} \leq 10$

Способ построения:

Закрашенная область состоит из цепочки квадратов размера 2×2 .
Цепочка квадратов начинается с правого нижнего угла поля.

Вариант 9



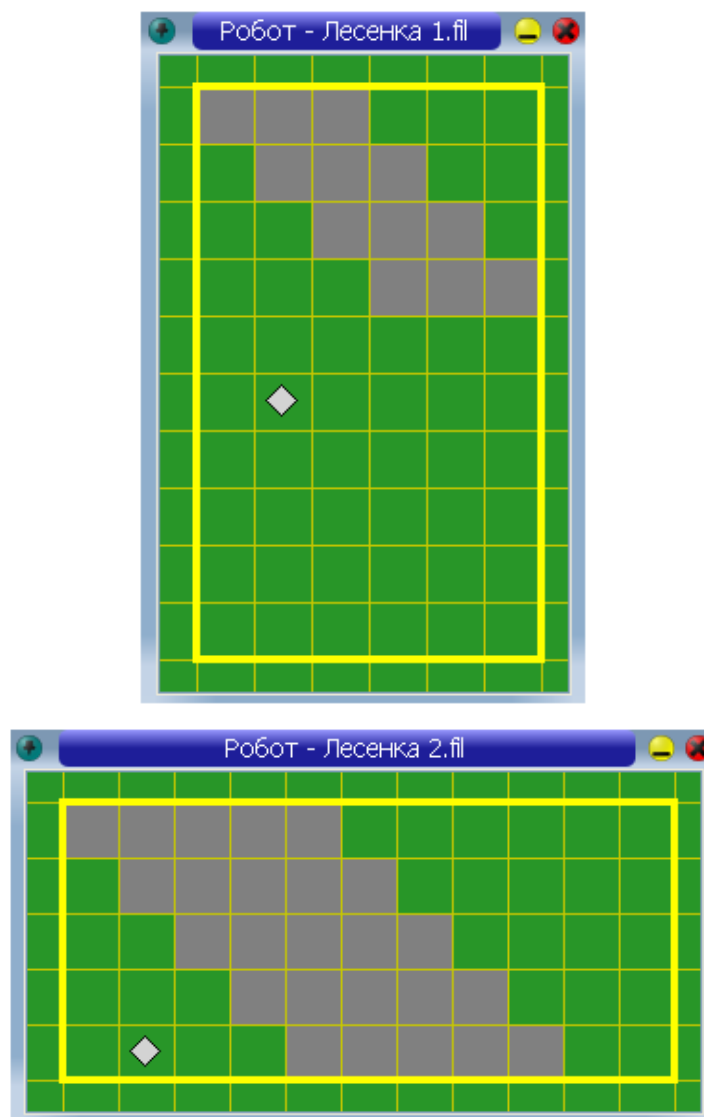
Размеры поля могут варьироваться в пределах:

$3 \leq \text{ширина} \leq 11$, $3 \leq \text{высота} \leq 10$

Способ построения:

Левый закрашенный столбец области равен высоте поля. Каждый следующий закрашенный столбец на две клетки меньше чем предыдущий столбец и отцентрирован по высоте поля.

Вариант 10



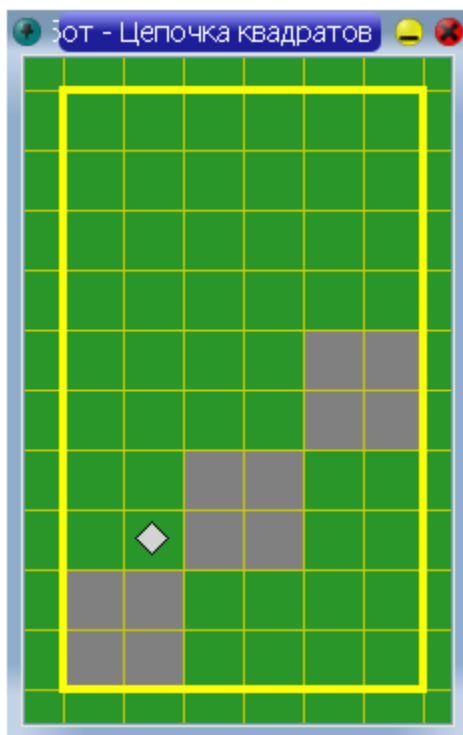
Размеры поля могут варьироваться в пределах:

$$4 \leq \text{ширина} \leq 11, 4 \leq \text{высота} \leq 10$$

Способ построения:

Верхняя закрашенная строка области равна (ширине поля)/2 (если получается дробное значение, то оно округляется вниз до целого значения) и привязана к левой границе поля. Каждая следующая закрашенная строка равна предыдущей строке, но смещена относительно нее на одну клетку вправо.

Вариант 11



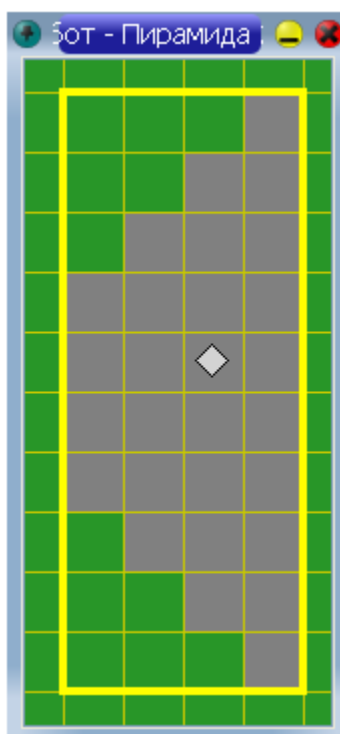
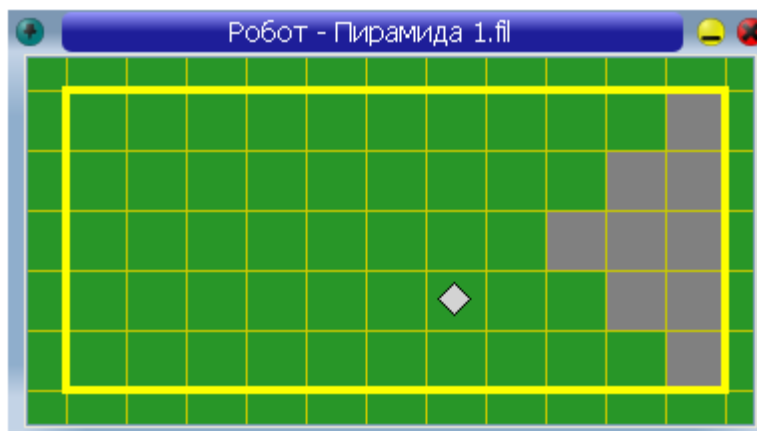
Размеры поля могут варьироваться в пределах:

$3 \leq \text{ширина} \leq 11$, $3 \leq \text{высота} \leq 10$

Способ построения:

Закрашенная область состоит из цепочки квадратов размера 2x2.
Цепочка квадратов начинается с левого нижнего угла поля.

Вариант 12



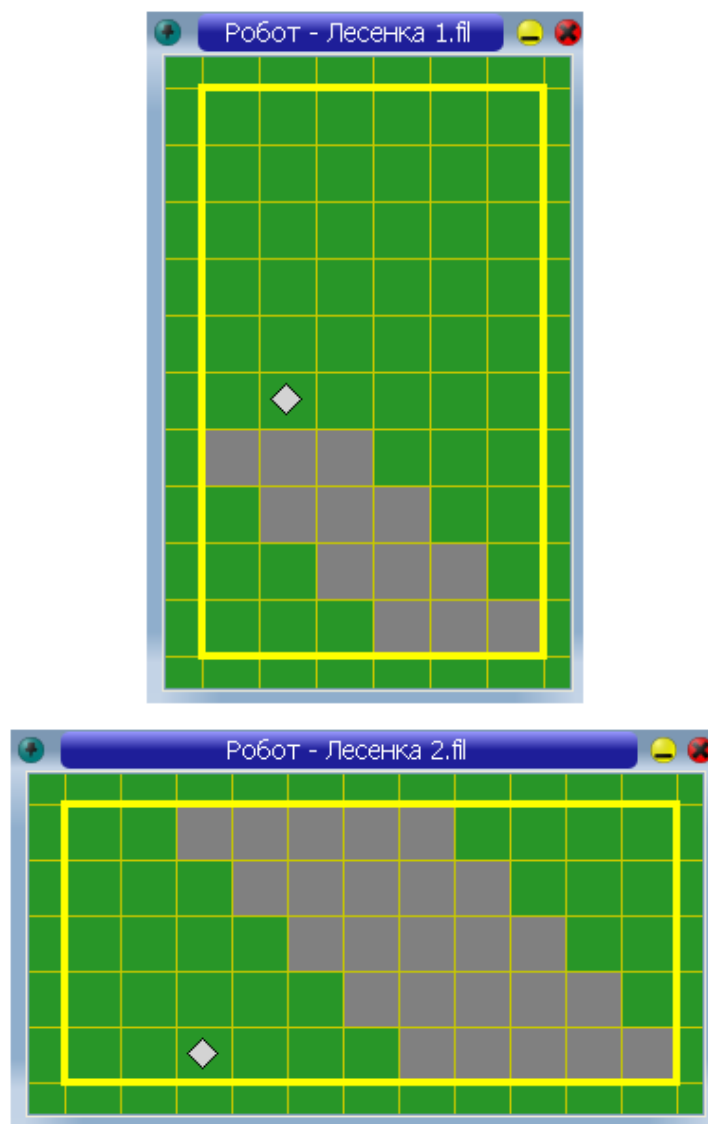
Размеры поля могут варьироваться в пределах:

$3 \leq \text{ширина} \leq 11$, $3 \leq \text{высота} \leq 10$

Способ построения:

Правый закрашенный столбец области равен высоте поля. Каждый следующий закрашенный столбец на две клетки меньше чем предыдущий столбец и отцентрирован по высоте поля.

Вариант 13



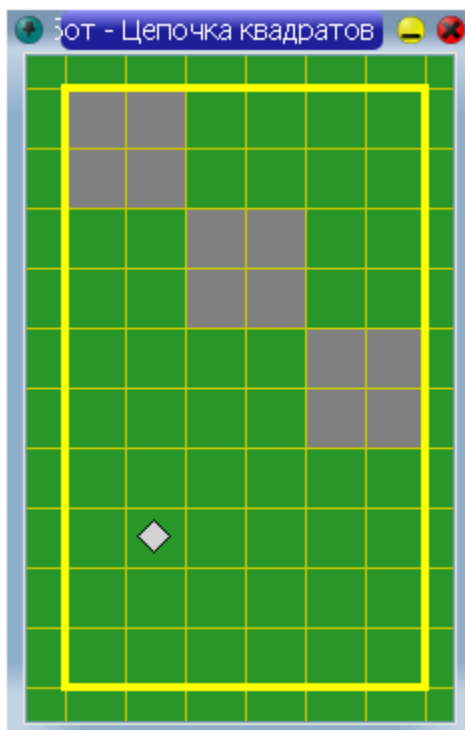
Размеры поля могут варьироваться в пределах:

$$4 \leq \text{ширина} \leq 11, 4 \leq \text{высота} \leq 10$$

Способ построения:

Нижняя закрашенная строка области равна (ширине поля)/2 (если получается дробное значение, то оно округляется вниз до целого значения) и привязана к правой границе поля. Каждая следующая закрашенная строка равна предыдущей строке, но смещена относительно нее на одну клетку влево.

Вариант 14



Размеры поля могут варьироваться в пределах:

$3 \leq \text{ширина} \leq 11$, $3 \leq \text{высота} \leq 10$

Способ построения:

Закрашенная область состоит из цепочки квадратов размера 2×2 .
Цепочка квадратов начинается с левого верхнего угла поля.

Задачи поиска источников радиации

Уровень 1

Вариант 1

Робот находится в **произвольной** клетке огороженного внешними стенами обстановки пространства, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны.** На поле часть клеток закрашена: **смотри свой вариант задания.**

В некоторых клетках **поля** может присутствовать радиация, но не обязательно. Необходимо **найти значение максимального источника радиации в закрашенной области.** Как результат напечатать сообщение "Максимальная радиация в закрашенной области равна max ", где max — максимальное значение радиации в закрашенной области.

Реализовать и использовать один из вспомогательных алгоритмов:

вещ макс_радиация_в_строке(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | максимальное значение радиации в строке

вещ макс_радиация_в_столбце(**лит** направление, **цел** длина)

дано | направление и длина столбца, в котором ищем радиацию

надо | максимальное значение радиации в столбце

вещ макс_радиация_по_диагонали(**лит** направление, **цел** длина)

дано | направление и длина диагонали, в которой ищем радиацию (указывается
| горизонтальное и вертикальное направление, например "вниз влево")

надо | максимальное значение радиации в диагонали

вещ макс_радиация_в_ряду(**лит** направление, **цел** длина)

дано | направление и длина ряда, в котором ищем радиацию

| (ряд - последовательность клеток, которая может быть горизонтальной
| или вертикальной)

надо | максимальное значение радиации в ряду

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закрашки в задании, не обращаясь к клеткам поля.

Вариант 2

Робот находится в **произвольной** клетке огороженного внешними стенами обстановки пространства, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны.** На поле часть клеток закрашена: **смотри свой вариант задания.**

В некоторых клетках **поля** может присутствовать радиация, но не обязательно. **Гарантируется наличие хотя бы одного источника радиации** в закрашенной области. Необходимо **определить среднеарифметическое значение радиации в закрашенной области** (клетки, не содержащие радиацию, не учитываются). Как результат напечатать сообщение "Среднеарифметическое значение радиации в закрашенной области равно avg", где avg — среднеарифметическое значение радиации в закрашенной области.

Реализовать и использовать один из вспомогательных алгоритмов:

вещ суммарная_радиация_в_строке(**лит** направление, **цел** длина, **рез цел** кол_во)

дано | направление и длина строки, в которой ищем радиацию

надо | суммарная радиация и кол-во источников радиации в строке

вещ суммарная_радиация_в_столбце(**лит** направление, **цел** длина, **рез цел** кол_во)

дано | направление и длина столбца, в котором ищем радиацию

надо | суммарная радиация и кол-во источников радиации в столбце

вещ суммарная_радиация_по_диагонали(**лит** направление, **цел** длина, **рез цел** кол_во)

дано | направление и длина диагонали, в которой ищем радиацию

| (направление - указывается горизонтальное и вертикальное

| направление, например "вниз влево")

надо | суммарная радиация и кол-во источников радиации на диагонали

вещ суммарная_радиация_в_ряду(**лит** направление, **цел** длина, **рез цел** кол_во)

дано | направление и длина ряда, в котором ищем радиацию

| (ряд - последовательность клеток, которая может быть горизонтальной

| или вертикальной)

надо | суммарная радиация и кол-во источников радиации в ряду

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закрашки в задании, не обращаясь к клеткам поля.

Вариант 3

Робот находится в **произвольной** клетке огороженного внешними стенами обстановки пространства, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны.** На поле часть клеток закрашена: **смотри свой вариант задания.**

В некоторых клетках **поля** может присутствовать радиация, но не обязательно. Необходимо **определить, что закрашенная область содержит только одинаковые источники радиации (ИР).** Как результат напечатать сообщения:

- "В закрашенной области **нет источников** радиации";
- "В закрашенной области **только один источник** радиации";
- "В закрашенной области **все источники** радиации **одинаковые**";
- "В закрашенной области **не все источники** радиации **одинаковые**".

Реализовать и использовать один из вспомогательных алгоритмов:

лог одинаковые_ИР_в_строке(**лит** направление, **цел** длина, **рез** **вещ** рад)

дано | направление и длина строки, в которой ищем радиацию

надо | признак того, что все источники радиации в строке одинаковые;

| рад - значение первого источника радиации

лог одинаковые_ИР_в_столбце(**лит** направление, **цел** длина, **рез** **вещ** рад)

дано | направление и длина столбца, в котором ищем радиацию

надо | признак того, что все источники радиации в столбце одинаковые;

| рад - значение первого источника радиации

лог одинаковые_ИР_по_диагонали(**лит** направление, **цел** длина, **рез** **вещ** рад)

дано | направление и длина диагонали, в которой ищем радиацию

| (направление - указывается горизонтальное и вертикальное

| направление, например "вниз влево")

надо | признак того, что все источники радиации одинаковые;

| рад - значение первого источника радиации

лог одинаковые_ИР_в_ряду(**лит** направление, **цел** длина, **рез** вещь рад)

дано | направление и длина ряда, в котором ищем радиацию

| (ряд - последовательность клеток, которая может быть горизонтальной
| или вертикальной)

надо | признак того, что все источники радиации одинаковые;

| рад - значение первого источника радиации

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закрашки в задании, не обращаясь к клеткам поля.

Вариант 4

Робот находится в **произвольной клетке закрашенной области внутри огороженного пространства**, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны**. Форма закрашенной области: **смотри в своем варианте задания**.

В некоторых клетках **поля** может присутствовать радиация, но не обязательно. Необходимо **определить ближайший к первоначальной позиции робота источник радиации в закрашенной области**. Как результат — поставить робота в эту клетку. Дополнительно напечатать сообщения:

- "Радиация **обнаружена** в закрашенной области";
- "Радиация **НЕ обнаружена** в закрашенной области".

Реализовать и использовать один из вспомогательных алгоритмов:

цел найти_ИР_в_строке(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | робот в клетке с радиацией и/или последней клетке строки; кол-во шагов, пройденных до источника радиации

цел найти_ИР_в_столбце(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | робот в клетке с радиацией и/или последней клетке строки; кол-во

шагов, пройденных до источника радиации

цел найти_ИР_по_диагонали(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | робот в клетке с радиацией и/или последней клетке строки; кол-во шагов, пройденных до источника радиации

цел найти_ИР_в_ряду(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | робот в клетке с радиацией и/или последней клетке строки; кол-во шагов, пройденных до источника радиации

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закраски в задании, не обращаясь к клеткам поля.

Вариант 5

Робот находится в **левой верхней клетке** огороженного пространства, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны**. На поле часть клеток закрашено: **смотри свой вариант задания**.

В некоторых клетках **поля** может присутствовать радиация, но не обязательно. Необходимо **определить, имеется ли радиация в НЕ закрашенной области**. Как результат поставить робота в любую НЕ закрашенную клетку, содержащую радиацию. Дополнительно напечатать сообщения:

- "Радиация **обнаружена** в **незакрашенной** области";
- "Радиация **НЕ обнаружена** в **незакрашенной** области".

Реализовать и использовать один из вспомогательных алгоритмов:

найти_ИР_в_строке(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | робот в клетке с радиацией и/или последней клетке строки

найти_ИР_в_столбце(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию
надо | робот в клетке с радиацией и/или последней клетке строки

найти_ИР_по_диагонали(**лит** направление, **цел** длина)
дано | направление и длина строки, в которой ищем радиацию
надо | робот в клетке с радиацией и/или последней клетке строки

найти_ИР_в_ряду(**лит** направление, **цел** длина)
дано | направление и длина строки, в которой ищем радиацию
надо | робот в клетке с радиацией и/или последней клетке строки

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закрашки в задании, не обращаясь к клеткам поля.

Уровень 2

Вариант 6

Робот находится в **левой верхней клетке** огороженного пространства, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны**. На поле часть клеток закрашено: **смотри свой вариант задания**.

В некоторых клетках **периметра** закрашенной области может присутствовать радиация, но не обязательно. Необходимо **определить, что имеется траектория движения робота из левого верхнего угла в правый нижний, в которой не содержится радиоактивных клеток**. Траектория может быть только из горизонтальных или вертикальных линий, но НЕ диагональных. Как результат напечатать сообщения:

- "Безопасная траектория **найдена**";
- "Безопасная траектория **НЕ найдена**".

Реализовать и использовать один из вспомогательных алгоритмов:

цел кол_во_ИР_в_строке(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | кол-во источников радиации

цел кол_во_ИР_в_столбце(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | кол-во источников радиации

цел кол_во_ИР_по_диагонали(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | кол-во источников радиации

цел кол_во_ИР_в_ряду(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | кол-во источников радиации

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закрашки

в задании, не обращаясь к клеткам поля.

Вариант 7

Робот находится в **левой верхней** клетке огороженного пространства, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны.** На поле часть клеток закрашено: **смотри свой вариант задания.**

В поле имеется **одно неразрывное радиоактивное пятно произвольных размеров.** Необходимо **определить, что пятно полностью находится в закрашенной области.** Как результат напечатать сообщения:

- "Радиационное пятно **внутри** закрашенной области";
- "Радиационное пятно **вне** закрашенной области";
- "Радиационное пятно **пересекается** с закрашенной областью".

Реализовать и использовать один из вспомогательных алгоритмов:

лог найти_ИР_в_строке(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | признак наличия радиации в строке, робот в последней клетке строки

лог найти_ИР_в_столбце(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | признак наличия радиации в строке, робот в последней клетке строки

лог найти_ИР_по_диагонали(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | признак наличия радиации в строке, робот в последней клетке строки

лог найти_ИР_в_ряду(**лит** направление, **цел** длина)

дано | направление и длина строки, в которой ищем радиацию

надо | признак наличия радиации в строке, робот в последней клетке строки

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закраски в задании, не обращаясь к клеткам поля.

Вариант 8

Робот находится в **произвольной** клетке огороженного внешними стенами обстановки пространства, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны.** На поле часть клеток закрашена: **смотри свой вариант задания.**

В некоторых клетках **поля** может присутствовать радиация, но не обязательно. Необходимо **найти расположение квадратной области размером 2x2, в которой суммарное значение радиации в клетках минимально.**

Для вариантов закрашенной области: 4, 8, 11, 14 искать **ВНЕ** закрашенной области.

Для остальных вариантов закрашенной области искать **ВНУТРИ** закрашенной области.

Как результат поставить робота в верхнюю левую клетку данной области и напечатать сообщение: “Уровень радиации в найденной области: rad”, где rad — суммарный уровень радиации всех клеток данной области.

Если радиации в поле нет, оставить робота в произвольном месте, напечатать сообщение: “Радиация в поле не найдена”.

Реализовать и использовать один из вспомогательных алгоритмов:

лог суммарная_радиация_в_квадрате(**лит** положение_робота, **рез** **вещ** рад)

дано | робот стоит в одной из клеток квадрата, положение_робота указывает, в
| какой клетке квадрата находится робот (например “слева сверху”)

надо | уместится ли квадрат в данном направлении
| рад - суммарная радиация в квадрате

вещ мин_радиация_в_ряду(**лит** направление, **цел** длина)

дано | направление и длина ряда, в котором ищем радиацию
| (ряд - последовательность клеток, которая может быть горизонтальной
| или вертикальной)

надо | минимальное значение радиации в ряду

лог суммарная_радиация_в_текущей_и_соседней_клетках(**лит** направление, **рез** **вещ** рад)

дано | направление, в котором ищем, где находится соседняя клетка
| направление может быть как прямое, так и по диагонали,
| например “слева сверху”)

надо | есть ли рядом соседняя клетка в указанном направлении (не стена)
| суммарное значение радиации в клетке, в которой находится робот, и в
| соседней клетке

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закраски в задании, не обращаясь к клеткам поля.

Вариант 9

Робот находится в **произвольной** клетке огороженного внешними стенами обстановки пространства, имеющего форму прямоугольника. **Размеры прямоугольника неизвестны**. На поле часть клеток закрашена: **смотри свой вариант задания**.

В некоторых клетках **поля** может присутствовать радиация, но не обязательно. Необходимо **найти такую пару граничащих клеток (закрашенная-чистая), в которой разность значений радиации в этих клетках будет максимальна**. Клетки, стоящие друг от друга на диагонали не рассматриваются. Как результат поставить робота на закрашенную клетку найденной пары клеток и напечатать сообщение: "Соседняя клетка находится: direction, разность значений радиации клеток: rad", где direction — направление, в котором находится соседняя не закрашенная клетка, rad — разность значений радиации данной пары клеток.

Если существует две и более клеток, которые граничат с одной и той же клеткой и в этих клетках разность уровней радиации одинакова и максимальна, то **выбрать любую из этих пар клеток** (например, в случае, если закрашенные клетки с одинаковым уровнем радиации окружают незакрашенную и разность значений радиации максимальна).

Если во всех граничащих клетках (закрашенная-чистая) радиации нет, то напечатать сообщение: "В граничащих клетках радиация не обнаружена".

Реализовать и использовать один из вспомогательных алгоритмов:

лог радиация_на_границе_строки(**рез** **вещ** рад)

дано | робот находится в строке-последовательности клеток одного типа
| (закрашенных или нет)

надо | есть ли в данной строке переход с одного типа клеток на другой
| рад - значение радиации в граничащей клетке текущего типа

лог радиация_на_границе_столбца(**рез** **вещ** рад)

дано | робот находится в столбце-последовательности клеток одного типа
| (закрашенных или нет)

надо | есть ли в данном столбце переход с одного типа клеток на другой
| рад - значение радиации в граничной клетке текущего типа

лог радиация_на_границе_ряда(**лит** направление, **рез** **вещ** рад)

дано | робот находится в ряду-последовательности клеток одного типа
| (закрашенных или нет), направление ряда. Ряд - последовательность
| клеток, которая может быть горизонтальной или вертикальной

надо | есть ли в данном ряду переход с одного типа клеток на другой
| рад - значение радиации в граничной клетке текущего типа

лог разность_значений_радиации(**лит** направление_к_второй_клетке, **рез** **вещ** разность)

дано | направление_к_второй_клетке - направление в котором рассматриваем
соседнюю
| клетку

надо | есть ли в указанном направлении клетка (не стена)
| разность - разность между значением радиации в текущей клетке и в
| соседней клетке

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно, закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закраски в задании, не обращаясь к клеткам поля.

Вариант 10

Робот находится в **произвольной** клетке **закрашенной области** **внутри** **огороженного пространства**, имеющего форму прямоугольника. **Размеры** **прямоугольника** **неизвестны**. Форма закрашенной области: **смотри в своем варианте задания**.

В некоторых клетках **поля** может присутствовать радиация, но не обязательно. Необходимо **определить источники радиации, испускающие излучение меньше некоторого значения Φ в закрашенной области**. Как результат вернуть количество таких источников. Значение Φ вводится с клавиатуры в начале работы программы. Дополнительно напечатать сообщения, информирующие о найденных источниках (вместо содержимого угловых скобок необходимо вставить в строку соответствующее значение):

- "Радиация ниже значения <Значение Φ > в <Количество найденных источников> источник/а/ов"

ИЛИ

- "Радиация ниже значения <Значение Φ > не найдена!".

Реализовать и использовать один из вспомогательных алгоритмов (ИР - источник радиации):

цел счётчик_ИР_меньше_ Φ _в_строке(**лит** направление, **цел** длина, **вещ** Φ)

дано |направление, длина строки, в которой ищем радиацию, значение Φ - меньше которого ищем источники радиации

надо |робот на последней клетке строки; кол-во источников радиации, со значением, меньшим, чем Φ

цел счётчик_ИР_меньше_ Φ _в_столбце(**лит** направление, **цел** длина, **вещ** Φ)

дано |направление, длина строки, в которой ищем радиацию, значение Φ - меньше которого ищем источники радиации

надо |робот на последней клетке строки; кол-во источников радиации, со значением, меньшим, чем Φ

цел счётчик_ИР_меньше_ Φ _по_диагонали(**лит** направление, **цел** длина, **вещ** Φ)

дано |направление, длина строки, в которой ищем радиацию, значение Φ - меньше которого ищем источники радиации

надо |робот на последней клетке строки; кол-во источников радиации, со значением, меньшим, чем Φ

цел счётчик_ИР_меньше_ Φ _в_ряду(**лит** направление, **цел** длина, **вещ** Φ)

дано |направление, длина строки, в которой ищем радиацию, значение Φ - меньше которого ищем источники радиации

надо |робот на последней клетке строки; кол-во источников радиации, со значением, меньшим, чем Φ

Считается, что робот **НЕ МОЖЕТ** определить самостоятельно,

закрашена клетка или нет, т.е. **ЗАПРЕЩЕНО** использовать команды робота "**клетка закрашена**" и "**клетка чистая**". Расположение закрашенных клеток Робот должен определять исходя из правил закрашки в задании, не обращаясь к клеткам поля.

Варианты заданий

Уровень 1

Варианты 1 - 19

№	Задача поиска	Вид закрашенной области
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	1	6
7	1	7
8	1	8
9	1	9
10	1	10
11	1	11
12	1	12
13	1	13
14	1	14
15	2	1
16	2	2
17	2	3
18	2	4
19	2	5

Варианты 20 - 39

№	Задача поиска	Вид закрашенной области
20	2	6
21	2	7
22	2	8
23	2	9
24	2	10
25	2	11
26	2	12
27	2	13
28	2	14
29	3	1
30	3	2
31	3	3
32	3	4
33	3	5
34	3	6
35	3	7
36	3	8
37	3	9
38	3	10
39	3	11

Варианты 40 - 56

№	Задача поиска	Вид закрашенной области
40	3	12
41	3	13
42	3	14
43	4	1
44	4	2
45	4	3
46	4	4
47	4	5
48	4	6
49	4	7
50	4	8
51	4	9
52	4	10
53	4	11
54	4	12
55	4	13
56	4	14

Уровень 2

Варианты 57 - 76

№	Задача поиска	Вид закрашенной области
57	5	1
58	5	2
59	5	3
60	5	4
61	5	5
62	5	6
63	5	7
64	5	8
65	5	9
66	5	10
67	5	11
68	5	12
69	5	13
70	5	14
71	6	1
72	6	2
73	6	3
74	6	4
75	6	5
76	6	6

Варианты 77 - 96

№	Задача поиска	Вид закрашенной области
77	6	7
78	6	8
79	6	9
80	6	10
81	6	11
82	6	12
83	6	13
84	6	14
85	7	1
86	7	2
87	7	3
88	7	4
89	7	5
90	7	6
91	7	7
92	7	8
93	7	9
94	7	10
95	7	11
96	7	12

Варианты 97 - 116

№	Задача поиска	Вид закрашенной области
97	7	13
98	7	14
99	8	1
100	8	2
101	8	3
102	8	4
103	8	5
104	8	6
105	8	7
106	8	8
107	8	9
108	8	10
109	8	11
110	8	12
111	8	13
112	8	14
113	9	1
114	9	2
115	9	3
116	9	4

Варианты 117 - 140

№	Задача поиска	Вид закрашенной области
117	9	5
118	9	6
119	9	7
120	9	8
121	9	9
122	9	10
123	9	11
124	9	12
125	9	13
126	9	14
127	10	1
128	10	2
129	10	3
130	10	4
131	10	5
132	10	6
133	10	7
134	10	8
135	10	9
136	10	10
137	10	11
138	10	12
139	10	13
140	10	14