

Алгоритмы и лабиринты. Решения.

А.Я.Белов-Канель, И.А.Гажур, И.А.Иванов-Погодаев, А.С.Малистов

A11. Приведем решение для случая, когда роботы видят флажки друг друга. Пусть каждый робот расталкивает свои флажки по стандартному алгоритму обхода ленты, но первый – по вертикали, а второй по горизонтали. Кроме того, если робот увидел чужой флажок, то он останавливается и ждет. Пусть вертикаль и горизонталь пересекаются в точке O . Если роботы в начале находятся на одинаковом расстоянии от O , они придут в нее одновременно и встретятся. Пусть, теперь, для определенности, робот A в начале находится на большем расстоянии от O чем робот B . В этом случае, в момент прихода робота A в O , он будет находиться между двумя чужими флажками на ленте, по которой ходит B . В соответствии с алгоритмом обхода, A поставит в клетку O флажок, а учитывая, что роботы начали одновременно и следуют одному и тому же алгоритму обхода, за то время пока A добежит до другого своего флажка и вернется назад, робот B проделает путь от одного своего флажка до другого и, следовательно, пройдет через клетку O . Таким образом, он увидит чужой флаг и остановится. Через какое-то время к нему подойдет второй робот.

Задачу также можно решить для случая, когда роботы не видят чужие флаги. Ее решение оставляем читателю.

A12. В трехмерном пространстве Роботы уже могут не пересечься.

Докажем это. Нам потребуется следующее утверждение, описывающее систему “Робот– 2 флажка”:

Лемма. Дана система “Робот– 2 флажка” в n -мерном пространстве. Тогда имеют место следующие возможности:

- 1. Расстояние между флажками не стремится к бесконечности. Расстояние между Роботом и флажками не стремится к бесконечности. Система Робот-Флажки движется также, как Робот без флажков (возможно, с большим числом состояний, чем исходный Робот – прямолинейно и равномерно, вдоль прямой L , параллельной некоторому рациональному вектору. При этом расстояние от Робота до L ограничено.*
- 2. Расстояние между флажками не стремится к бесконечности. Расстояние между Роботом и флажками стремится к бесконечности. Робот движется прямолинейно и равномерно, удаляясь от флажков вдоль прямой L , параллельной рациональному вектору (координаты которого не превосходят числа внутренних состояний Робота). При этом расстояние от Робота до L ограничено.*
- 3. Расстояние между флажками стремится к бесконечности. Робот только конечное число раз встречает один флажок F_1 . Робот движется прямолинейно и равномерно, удаляясь от F_1 вдоль прямой L , параллельной некоторому рациональному вектору. При этом расстояние от Робота и обеих флажков до L ограничено.*
- 4. Расстояние между флажками стремится к бесконечности. Робот бесконечное число раз встречает оба флажка F_i . В этом случае существует такая константа D вектор \vec{e} , координаты которого не превосходят числа внутренних состояний Робота, прямая $L(T)$, параллельная \vec{e} , такая, что:*
 - Робот и флажки находятся от прямой $L(t)$ на расстоянии, не превосходящем D .*
 - Робот движется прямолинейно и равномерно между двумя флажками вдоль \vec{e} , отражаясь от одного флажка и возвращаясь к другому.*
 - Типов таких движений между соударениями конечное число, и эти типы периодически повторяются.*
 - Существует пара векторов $\vec{\lambda} = (\vec{\lambda}_1, \dots, \vec{\lambda}_n)$ и $\vec{\delta} = (\vec{\delta}_1, \dots, \vec{\delta}_n)$ и константа E такие, что прямая $L(t)$, Робот находится на расстоянии, не превосходящем E от отрезка $[\vec{\lambda}\sqrt{t}, \vec{\delta}\sqrt{t}]$, флажок F_1 находится на расстоянии, не превосходящем E от $\vec{\lambda}\sqrt{t}$, флажок F_1 находится на расстоянии, не превосходящем E от $\vec{\delta}\sqrt{t}$.*

Перейдем к **решению задачи A12**. Построим фазовую диаграмму пространство-время, введем ось T и рассмотрим мировые линии Робота и флажков. Переформулируем нашу лемму.

Дана система “Робот– 2 флажка” в n -мерном пространстве-времени. Тогда существует константа G такая, что имеют место следующие возможности

- 1. В ситуации 1) мировые линии Робота и флажков находятся на расстоянии $< G$ от некоторой прямой пространства-времени.*

2. В ситуациях 2)–3) мировые линии Робота и флажков находятся на расстоянии $< G$ от некоторой пары прямых пространства-времени, при этом одна из них задается условием “пространственная координата – константа”.
3. В ситуации 4) Робот и флажки находятся на расстоянии $< G$ от поверхности, состоящей из точек вида $\{(X, T) | X \in [\bar{\lambda}\sqrt{t}, \bar{\delta}\sqrt{t}]\}$.

Решение задачи A12 вытекает из такого наблюдения: любые два множества, описанные в предыдущей лемме, можно так сдвинуть вдоль вектора, перпендикулярного оси времени, чтобы они не пересеклись.

A13. Нет, это не верно. В качестве контрпримера двух лабиринтов можно взять две полуленты. Для них существуют очевидные программы обхода: для ленты бесконечной направо алгоритм обхода следующий: «двигаться налево пока не найдешь стенку впереди, после этого двигаться направо». Аналогичный алгоритм есть для бесконечной налево полуленты.

Допустим, существует универсальный алгоритм для обеих полулент. Запустим его на бесконечной в обе стороны ленте. В какой-то момент состояния повторятся. После этого робот будет удаляться либо вправо по ленте, либо влево. Значит, выбрать такую полуленту, бесконечную только в одну сторону, что она накроет все клетки, где когда либо был робот, плюс еще несколько клеток где он никогда не будет. При обходе этой полуленты робот не сможет отличить ее от полной ленты и поэтому обойти всю не сможет.

A14. Решение этой задачи разбирается в прилагаемой статье Даниила Гусева (файл gusev.pdf).

A15. Решение этой задачи жюри пока не известно.

Циклы В и С.

Решениями задач в циклах В и С являются алгоритмы-схемы, которые можно запускать в специальном приложении VillagerLife (оно прилагается).

В прилагаемых материалах находятся папки, по одной на каждую задачу. В них находятся несколько тестовых карт (названия test1.vlm, test2.vlm и т.д.), а также алгоритмы, решающие данную задачу. Мы приводим алгоритмы, которые написали участники конференции, их имена указаны в названии файла. Если имен не указано, алгоритм был составлен жюри.

Интересные алгоритмы для приведенных задач составили Дмитрий Беликов и Константин Хадаев (совместно), Данила Байгушев, и Брайан Чен, который является единственным участником, решившим сложную задачу C10.

Задачи C5-C7 фактически являются служебными к C8, поэтому тесты и программы для них не приведены.

Для задачи C8 есть два варианта тестов: против двух AI программ. Также приведен алгоритм, составленный Дмитрием Беликовым и Константином Хадаевым.