

Содержание

Введение	5
Основная часть	7
Формат	7
Используемые структуры данных	10

Введение

Самолёты крепко обосновались в нашей жизни. Уже в середине прошлого века самолёт не был чем-то необычным. А время шло. И вот уже миллионы людей летают ежемесячно по всему миру на самых разнообразных воздушных лайнерах.

Но самолёты не автомобили – они намного массивнее и много менее поворотливы. А самое главное отличие состоит в том, что во время полёта они не могут остановиться в случае опасности, когда для автомобиля это один из самых безопасных способов разрешить сложную ситуацию. Это всё не доставляло особых проблем, пока самолёты были явлением редким, экспериментальным. Но в связи с постоянно растущим авиапарком у компаний возникла потребность в регулировке движения этих мастодонтов. Поэтому оказались чрезвычайно важными грамотная и своевременная дача указаний к движению воздушных бортов. У человечества к тому времени был обширный опыт регулирования наземного и водного транспорта, поэтому для разработки новых правил движения широко использовались эти наработки. Но всё равно воздух это другая стихия. И помимо стандартных решений нужны были новые.

Если на трассе между городами всё было более-менее понятно и решения имеющихся проблем нашлись довольно быстро. А вот вблизи больших городов, когда в аэропорт прилетают самолёты отовсюду, надо было принимать серьёзные меры.

Самым первым, и использующимся по сей день, стало расписание полётов. Но расписание это разграничения в рамках часов, а когда к аэропорту подлетают самолёты для посадки, нужны решения с точностью до минут, а лучше – десятков секунд. Диспетчер – человек, ответственный за такое распределение. В помощь ему создаются компьютерные программы, которые рассчитывают примерное время прибытия воздушных судов, и диспетчер, используя эти данные, может лучше регулировать входящий поток.

Постановка задачи

Разработать численную процедуру, которая будет рассчитывать возможные интервалы прибытия для каждой контрольной точки на пути следования самолёта.

Условности и входные данные

Будем считать, что самолёт это материальная точка,двигающаяся равнопеременно и в начальной точке потока временной интервал нулевой. Необходимы три входных файла, соответствующих формату:

1. Файл с данными контрольных точек
2. Файл с данными о схемах
3. Файл с данными о потоках

Работа программы

Программа строит граф всей воздушной зоны, на основании схем и потоков. Затем для каждого потока строит подграф, который топологически сортирует и вычисляет возможные временные интервалы прибытия в каждую точку, считая, что в начальной точке потока временной интервал – нулевой.

Основная часть

Формат

Определение 1. *Спрявление* - возможность прервать выполнение текущей схемы и уйти на заданную(ые) точки.

Имеет следующий вид:

Str(Точка1 Точка2 ...ТочкаK) Точка1 Точка2 ...ТочкаK /Str

Str(Точка1 Точка2 ...ТочкаK) - указание на какие точки возможно спрявление, а пункты, заключённые между Str() и /Str - с каких точек возможно спрявление

Точки разделены пробелами.

Определение 2. *Схема* - последовательность точек.

Считаем, что схемы пересекаются только по начальной и конечным точкам. В одной схеме может быть только одно спрявление.

Определение 3. *Начальная точка схемы* - первая точка этой схемы.

Определение 4. *Конечная(ые) точка(ы) схемы* - конечными точками схемы считаются ее последняя точка, а также не принадлежащие ей точки, на которые возможно спрявление.

Определение 5. *Стандартная схема* - схема, описываемая тремя точками и имеющая определённый вид.

РИСУНОК СТАНДАРТНОЙ СХЕМЫ

Будем считать, что стандартную схему можно проходить не более заданного числа раз.

Определение 6. *Поток* - последовательность схем без циклов, имеющая точку начала и оканчивающаяся схемой, которая содержит точку с флагом 'LAND'.

Файл с данными точек

В первой строке указывается количество точек - число из \mathbb{N}

Далее перечисляются контрольные точки в соответствии следующему формату:

Имя начинается с буквы. Каждая точка в новой строке. Между элементами - пробел.

Название x y z V_{min} V_{max} флаг посадочной полосы

$x, y, z \in \mathbb{R}$, где x и y - координаты в плоскости земли, а z - вертикальная составляющая. Допускается ввод в метрах, километрах и морских милях.

$V_{min}, V_{max} \in \mathbb{R}^+$, где V_{min} - минимально допустимая скорость, а V_{max} - максимально допустимая скорость на данном контрольном пункте. Допускается ввод в метрах в секунду, километрах в час и морских милях в час.

флаг посадочной полосы - 'LAND' иначе пусто или ноль

Примеры:

- SS025 0 0 900 70 90
- RW08L -21.5 18431.5 10 65 85 LAND

Файл с данными об обычных и стандартных схемах

В первой строке указывается количество схем - число из \mathbb{N}

Во второй строке указывается количество стандартных схем - число из \mathbb{N}

Далее перечисляются все схемы в формате:

Название схемы - любой набор букв и цифр. Точки отделяются пробелами. После двоеточия начинать с начальной точки.

Название схемы (точка начала схемы)(точки конца схемы) : название точек
[спрямление] название точек

Формат для стандартных схем:

Название стандартной схемы (точка начала стандартной схемы)(количество повторений стандартной схемы) : точка начала точка конца разворота точка конца
обратного плеча

Некоторый "синтаксический сахар":

1. В стандартной схеме можно опустить '(точка начала стандартной схемы)' и написать: Имя (количество повторений): T1 T2 T3
2. Если схема состоит только из двух точек: 'NameTwoP (A)(B): A B', то можно опустить точки после двоеточия: 'NameTwoP (A)(B):'

Примеры:

- NameA (DIPOP)(EE500 KOLOS): Str(KOLOS) DIPOP EE500 /Str

- NameB (KOLOS)(BEKAR) : KOLOS VALET Str(RODEL) EE020 EE021 EE022 EE023 EE024 EE025 /Str RODEL BEKAR
- NameC (RODEL)(RW25R): RODEL BEKAR KVOTA EE252 EE253 RW25R
- NameStScheme (GALEB)(5): GALEB N1 N2

Файл с данными о потоках

В первой строчке количество потоков - число из \mathbb{N}

Далее каждый поток в новой строке, согласно следующему правилу:

Имя потока Имя первой точки этого потока

Пример: 2

Flow1 DIPOP

Flow2 AKERA

Используемые структуры данных