

Система Навигации И Поиска Пути

Краткое Руководство Пользователя

Шаг 1.

- Импортируйте пакет СНИИП в текущий проект

Для того, чтобы начать работу с СНИИП, просто импортируйте пакет с СНИИП в ваш текущий проект. После импорта редактор СНИИП станет доступен в меню «Window» главного меню редактора Unity.

Шаг 2.

- Откройте игровой уровень с назначенной моделью коллизии

- Создайте новый файл навигации

- Добавьте в файл новый тип пути

- Создайте узлы для этого типа пути

- Сохраните файл

Откройте сцену с игровым уровнем для которого вы хотите создать навигацию, и убедитесь, что для этого уровня создана модель коллизии. После этого откройте редактор СНИИП (Window\NAPS), выберите опцию создания нового файла, затем добавьте одну подсеть путей(или новый тип пути) и приступите к размещению узлов внутри этой подсети. Для того, чтобы понять как это сделать, следуйте инструкциям описанным в подразделах «Начало работы», «Визуализация и добавление новых типов путей», «Создание узлов пути» раздела «Работа с редактором СНИИП» руководства пользователя СНИИП. Также вы можете увидеть процесс создания навигации в видео «NAPS introduction».

В этом примере мы создадим навигацию для приложенного примера ИИ, который управляет роботом. Данный ИИ может только генерировать для себя маршруты и перемещаться по ним. Размещайте узлы созданной подсети путей на горизонтальных поверхностях, так как по умолчанию робот не умеет перемещаться по вертикальным поверхностям.

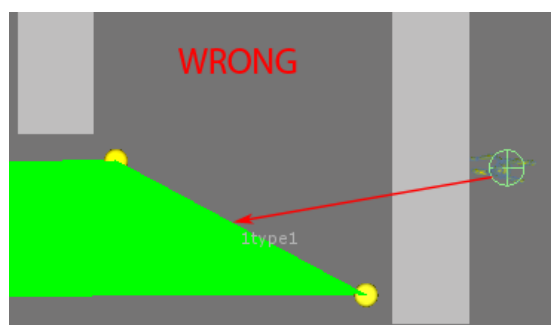
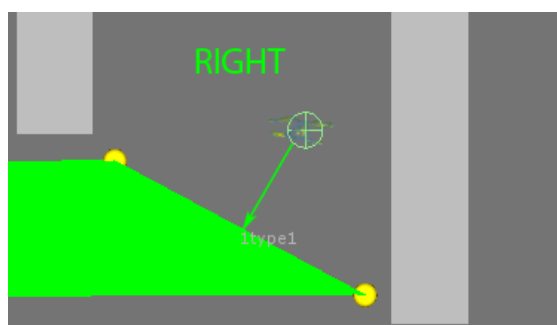
После того, как узлы будут размещены - сохраните файл.

Шаг 3.

- **Добавьте скрипт *PathfindingManager.cs* любому активному объекту в сцене и укажите в нем путь к файлу и его имя**
- **Перетащите префаб с NPC в сцену**
- **Разрешите данному NPC следовать по созданной подсети**

Скрипт *PathfindingManager.cs* нужен лишь для загрузки файлов навигации при запуске приложения. С помощью него вы также можете в реальном времени загружать нужные навигационные файлы.

После того, как подсеть путей была создана, можно позволить роботу начать двигаться используя ее данные. Для этого перетащите префаб с роботом под названием «Character», который находится в папке «prefabs\characters» в сцену, так чтобы робот не проваливался в коллизию игрового уровня и при этом находился внутри навигационной сетки, или по крайней мере, мог напрямую пройти к одному из ее узлов.

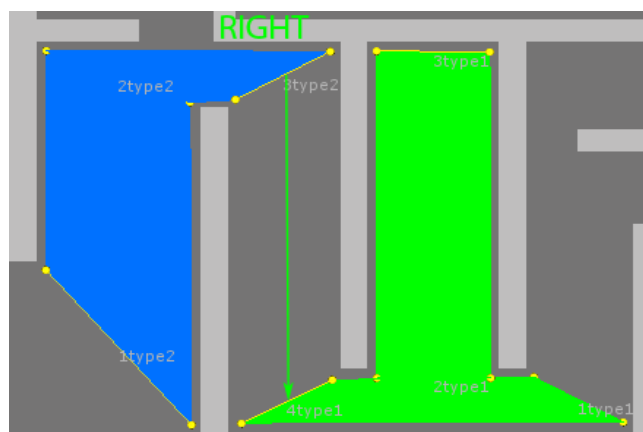
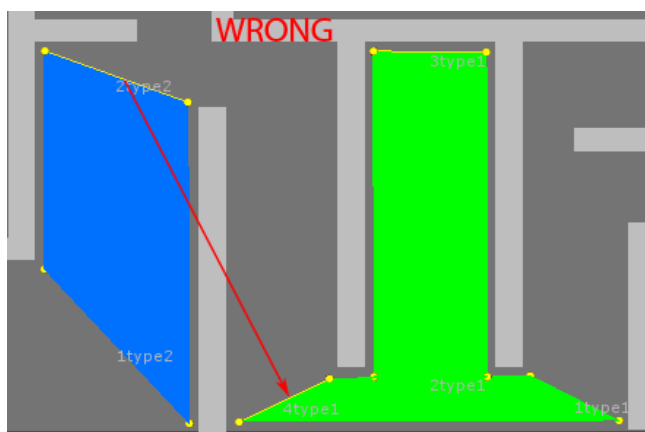


Затем выделите объект «Character» и в инспекторе в поле скрипта *PathFollowingAI* найдите переменную *Types*. Эта переменная представляет собой массив, установите его размер = 1 и в поле *element0* введите название созданной вами подсети путей. После этого, нажав кнопку «Play» в редакторе Unity вы обнаружите, что персонаж начал двигаться.

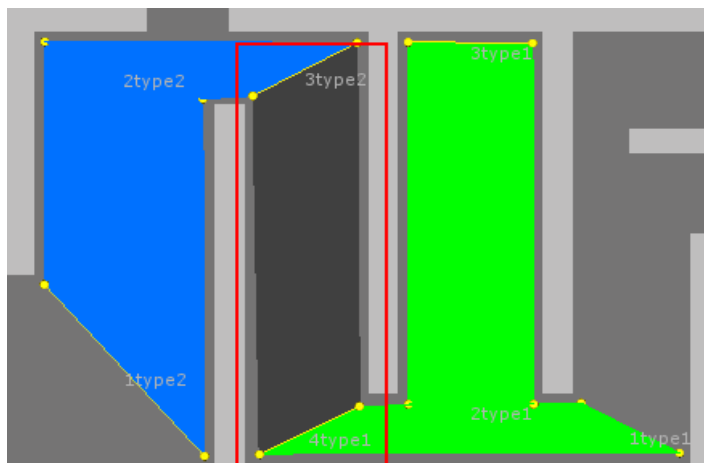
Шаг 4.

- **Добавьте в текущий файл еще одну подсеть путей**
- **Создайте для этой подсети узлы**
- **Добавьте в сцену еще одного NPC**
- **Разрешите данному NPC следовать по обеим подсетям**

Добавьте еще одну подсеть в текущий файл, точно также, как была добавлена предыдущая. То есть введите любое название в поле «New path» и нажмите кнопку «Add». Для того, чтобы начать размещение узлов в этой подсети, выберите ее с помощью кнопок со стрелками в поле «Current path type» и нажмите кнопку «Start placing». Разместите узлы этой подсети так, чтобы, по крайней мере, из одного ее узла был виден, по крайней мере, один узел предыдущей подсети.



Затем соедините эти два узла разных подсетей двухсторонней атипичной связью. Для того, чтобы понять как это сделать прочтите подразделы «Работа со связями узлов» и «Атипичные соединения» раздела «Работа с редактором СНИПП» руководства пользователя СНИПП. После создания двухстороннего атипичного соединения оно отобразится в редакторе темно серым цветом.



С помощью атипичного соединения мы объединили две отдельные подсети. Это значит, что теперь NPC смогут использовать эти навигационные подсети как по отдельности, так и как одну большую навигационную сеть.

Добавьте еще один экземпляр робота в сцену, можете просто скопировать уже имеющегося в сцене. Разместите второго робота рядом с предыдущим. Выделите его главный объект под названием «Character» и в поле скрипта PathFollowingAI установите параметр size переменной types равным 2. Затем в поле element0 напишите название первой подсети, в поле element1 напишите название второй подсети. После нажатия кнопки «Play» в редакторе Unity вы сможете заметить, что один из роботов генерирует маршруты проходящие через обе навигационные сетки, в то время как другой — генерирует маршруты только в одной подсети.

Шаг 5.

- Создайте новый тип пути для камеры используя одиночные точки

- Добавьте кривые между узлами этого пути

- Перетащите в сцену префаб CinematicCamera и задайте для нее маршрут для следования

Для создания путей для камеры, создайте новый путь указанным выше способом, разместив одиночные узлы в ключевых местах. Чтобы понять как создавать одиночные точки, прочтите подраздел «Создание узлов пути» раздела «Работа с редактором СНИИП» руководства пользователя СНИИП. После чего добавьте кривые между соседними узлами и задайте требуемый вид кривым, для более подробной информации об этом смотрите подраздел «Работа с кривыми» раздела «Работа с редактором СНИИП» руководства пользователя СНИИП. Затем перетащите в сцену префаб «Cinematic camera». Выделите этот объект в окне иерархии и в параметрах скрипта «Cinematic Camera» задайте размер переменной «Route», для этого установите значение size переменной Route равным количеству узлов в созданном вами новом пути. После этого заполните появившийся список элементов в переменной Route названиями узлов из созданного пути, в порядке, в котором они соединены кривыми.

Если все сделано правильно, то после нажатия кнопки «Play» камера начнет движение по созданному пути. Если созданный путь для камеры циклический, то есть последний и первый узлы маршрута соединяет кривая, то в поле скрипта «Cinematic Camera» вы можете установить флажок «Cycle route», в этом случае камера будет заиклленно двигаться по маршруту.

Примечание: Вы также можете добавлять кривые к паре соединенных

узлов любого пути и использовать такой путь как путь для камеры. Но если вы создаете специальный путь, который будет использоваться только кинематографической камерой или чем-то подобным, то не имеет смысла создавать для этой цели бипоинты, вместо этого используйте одиночные точки.

Более подробную информацию об использовании СНИПП вы можете получить ознакомившись с руководством пользователя СНИИП, а также изучив примеры исходных кодов.