

Beschreibung zur Wegfindung für Cargo Lagerplatz

Habe den Eintrag durch die Kommentare ergänzt Danke dafür

Also die Wegfindung wird in der *.mdl Datei der Ladebucht hinterlegt und diese befinden sich in diesem Ordner: res\models\model\station\road\
Ich werde jetzt den Code des Original cargo_medium_old.mdl hier versuchen zu Beschreiben was wofür ist.

Code

```
1. [code]local laneutil = require "laneutil"
2. function data()
5. return {
6. boundingInfo = {
7. bbMax = {
8. 23, 17, 5.2159490585327, /Breite, Tiefe, Höhe/ eines Objektes vom Nullpunkt aus positive Werte
9. },
10. bbMin = {
11. -23, -15, -0.013722999952734, /Breite, Tiefe, Höhe/ eines Objektes vom Nullpunkt aus negative Werte
12. Der bbMin Wert der Tiefe "-15" ist die Kante zur Straße
13. },
14. },
15. collider = {
16. params = {
17. },
18. type = "MESH",
19. },
20. lods = {
21. {
22. animations = {
23. },
24. children = { /Hier werden alle Pfade zu den Mashdateien angegeben/
25. {
26. id = "station/road/cargo_medium_old_lod_0_floor.msh", /Untergrund/
27. transf = {
28. 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, /Hier könnt ihr die Position des Objektes verändern alle
Angaben sind in Meter was jetzt genau für was ist weis ich grade leider nicht. Wenn ihr allerdings ein
Objekt Spiegeln wollt einfach auf Pos 1 und 6 ein minus davor/
29. },
30. type = "MESH",
31. }, {
32. id = "station/road/cargo_medium_old_lod_0_lanes.msh", /Untergrund/
33. transf = {
34. 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1,
35. },
36. type = "MESH",
37. }, {
38. id = "station/road/cargo_medium_old_lod_0_station.msh", /Gebäude/
```

```

51. transf = {
52. 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1,
53. },
54. type = "MESH",
55. }, {
58. id = "asset/industry/pallet_01_lod_0_pallet_01.msh", /Deko/
59. transf = {
60. 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, -2.8516, -4.5346, 0.2862, 1,
61. },
62. type = "MESH",
63. }, {
66. id = "asset/industry/pallet_01_lod_0_pallet_01.msh", /Deko/
67. transf = {
68. 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, -1.6516, -4.5346, 0.2862, 1,
69. },
70. type = "MESH",
71. }, {
74. id = "asset/industry/pallet_01_lod_0_pallet_01.msh", /Deko/
75. transf = {
76. 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, -1.6916, -4.5346, 0.432, 1,
77. },
78. type = "MESH",
79. }, {
82. id = "asset/industry/pallet_01_lod_0_pallet_01.msh", /Deko/
83. transf = {
84. 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, -0.5516, -4.5346, 0.4862, 1,
85. },
86. type = "MESH",
87. },
90. },
91. events = {
92. },
95. matConfigs = { /Hier müssen so viele 0, eingetragen werden wie oben *.msh Pfade Sonst Crasht das
    Game sofort/
96. {
97. 0, 0, 0,
98. 0, 0, 0,
99. 0,
100. },
101. },
102. static = false,
103. visibleFrom = 0, /Sichtbar von "0m"/
104. visibleTo = 5410, /bis "5410m" Entfernung/
105. },
108. },
109. metadata = {
110. description = {
111. name = _("Truck station"),
112. --description = _("Description")
113. },
114. availability = {
115. },
116. streetTerminal = {

```

117. cargo = true
 118. },
 119. streetConnector = {
 120. streetType = "old_medium.lua" /je nach dem welche *.lua hier angegeben wird ändert sich die max. Geschwindigkeit auf dem Platz/
 121. },
 122. transportNetworkProvider = {
 123. laneLists = {
 124. {
 125. transportModes = { "TRUCK" },
 126. nodes = laneutil.makeLanes({ /So nun zum eigentlichen Thema/
 127. -- lane1
 128. { { Seitlicher Versatz von 0, tiefen Versatz von 0, höhen Versatz von 0 }, /beim nachfolgendem Teil handelt es sich um die Kurven Radien{ 7.0, -10.0, .0 }, { .0, 7.85, .0 }, { 7.85, .0, .0 }, 3.0 },
 129. /hier habe ich mir jetzt einmal den ersten Dreierblock genommen und zur Veranschaulichung bearbeitet
 130. { { S1, -T1, H1 }, { S2, -T2, H2 }, { KS1, KT1, KH1 }, { KS2, KT2, KH2 }, 3.0 },
 133. { { S2, -T2, H2 }, { S3, -T3, H3 }, { KS2, KT2, KH2 }, { KS3, KT3, KH3 }, 3.0 },
 134. { { S3, -T3, H3 }, { S4, -T4, H4 }, { KS3, KT3, KH3 }, { KS4, KT4, KH4 }, 3.0 },
 136. /Man muss sich das wie folgt vorstellen
 138. { S1, -T1, H1 } ist der Startpunkt einer Linie { S2, -T2, H2 } ist der Endpunkt
 139. der Startpunkt der nächsten Linie muss mit dem Endpunkt der ersten übereinstimmen
 140. Genau das gleiche Pinzip ist auch bei den Kurven Radien
 141. Soweit mir bekannt steht die 3.0 am Ende für die Breite der Linie, sie wäre also 3m breit/
 142. /nun in Bezug auf die Original Daten
 143. { { 2.0, -15.0, .0 } dies ist der erste Punkt der Linie
 144. er ist Seitlich 2m verschoben und -15m in der Raumtiefe an dieser Stelle befindet sich der
 145. Anschluss zur Straße für die Einfahrt d.h. die Ausfahrt befindet sich auf { -2.0, -15.0, .0 }
 146. Einfahrt
 149. { { 2.0, -15.0, .0 }, { 7.0, -10.0, .0 }, { .0, 7.85, .0 }, { 7.85, .0, .0 }, 3.0 },
 150. { { 7.0, -10.0, .0 }, { 12.0, -5.0, .0 }, { 7.85, .0, .0 }, { .0, 7.85, .0 }, 3.0 },
 151. { { 12.0, -5.0, .0 }, { 12.0, 1.0, .0 }, { .0, 1.0, .0 }, { .0, 1.0, .0 }, 3.0 },
 152. { { 12.0, 1.0, .0 }, { 12.0, 7.0, .0 }, { .0, 1.0, .0 }, { .0, 1.0, .0 }, 3.0 },
 155. { { 12.0, 7.0, .0 }, { 7.0, 12.0, .0 }, { .0, 7.85, .0 }, { -7.85, .0, .0 }, 3.0 },
 156. { { 7.0, 12.0, .0 }, { 2.0, 12.0, .0 }, { -1.0, .0, .0 }, { -1.0, .0, .0 }, 3.0 },
 157. { { 2.0, 12.0, .0 }, { -2.0, 12.0, .0 }, { -1.0, .0, .0 }, { -1.0, .0, .0 }, 3.0 },
 158. { { -2.0, 12.0, .0 }, { -7.0, 12.0, .0 }, { -1.0, .0, .0 }, { -1.0, .0, .0 }, 3.0 },
 159. { { -7.0, 12.0, .0 }, { -12.0, 7.0, .0 }, { -7.85, .0, .0 }, { .0, -7.85, .0 }, 3.0 },
 160. { { -12.0, 7.0, .0 }, { -12.0, 1.0, .0 }, { .0, -1.0, .0 }, { .0, -1.0, .0 }, 3.0 },
 163. { { -12.0, 1.0, .0 }, { -12.0, -5.0, .0 }, { .0, -1.0, .0 }, { .0, -1.0, .0 }, 3.0 },
 166. { { -12.0, -5.0, .0 }, { -7.0, -10.0, .0 }, { .0, -7.85, .0 }, { 7.85, .0, .0 }, 3.0 },
 167. { { -7.0, -10.0, .0 }, { -2.0, -15.0, .0 }, { 7.85, .0, .0 }, { .0, -7.85, .0 }, 3.0 },
 168. Ausfahrt
 169. Alles was sich zwischen diesen beiden Werten befindet ist
 170. quasi eine Linie
 172. Bei den folgendenangaben handelt es sich um die Ausweichstrecken
 174. Startpunkte der Ausweiche
 175. { { 2.0, -15.0, .0 }, { 7.0, -5.0, .0 }, { 0, 1, .0 }, { .0, 1, .0 }, 3.0 },
 176. { { 7.0, -5.0, .0 }, { 7.0, 7.0, .0 }, { 0, 1, .0 }, { .0, 1, .0 }, 3.0 },
 177. { { 7.0, 7.0, .0 }, { 2.0, 12.0, .0 }, { .0, 7.85, .0 }, { -7.85, .0, .0 }, 3.0 },
 178. Endpunkte der Ausweiche
 179. { 2.0, 12.0, .0 } ist der Endpunkt der Ausweiche, dieser Punkt
 180. ist auch oben in der Hauptlinie zu finden.

```

181. Die Ausweiche dient dazu den Hin- bzw. Rückweg vom Halteplatz zur
182. Straße zu finden ohne dabei über einen anderen Halteplatz zu fahren
183. Startpunkt der Ausweiche
184. { { -2.0, 12.0, .0 }, { -7.0, 7.0, .0 }, { -7.85, .0, .0 }, { .0, -7.85, .0 }, 3.0 },
185. { { -7.0, 7.0, .0 }, { -7.0, -5.0, .0 }, { .0, -1, .0 }, { .0, -1, .0 }, 3.0 },
186. { { -7.0, -5.0, .0 }, { -2.0, -15.0, .0 }, { 0, -1, .0 }, { .0, -1, .0 }, 3.0 },
187. Endpunkt der Ausweiche
188. Demnach, wie auch im Spiel zu sehen, hat die kleine Cargo Ladestation eine Hauptlinie die
191. über beide Ladeplätze führt und zwei Ausweichen woraus sich zwei getrennte Linien ergeben.
192. ))
193. }, {
194. transportModes = { "CARGO" },
195. nodes = laneutil.makeLanes({ /Bei diesen Angaben handelt es sich um die Wegfindung für
196. das Cargo Overlay und gleichzeitig für die Abmaße des Bereichs der Lagerfläche
197. -- ped1
198. Straßenanschluss Bürgersteig
199. { { 6.5, -15.0, .3 }, { 14.0, -13.0, .3 }, { 0, 1, .0 }, { 1, 1, .0 }, 3.0 },
200. { { 14.0, -13.0, .3 }, { 18, -13.0, 1.0 }, { 1, 0, 1 }, { 1, 0, 1 }, 3.0 },
201. Startpunkt Lagerfläche /auf dieser höhe Stoppen auch die Fahrzeuge "siehe Nachtrag1 unten"
202. { { 18, -13.0, 1.0 }, { 18.0, -7.0, 1.0 }, { 0, 1, 0 }, { 0, 1, 0 }, 3.0 },
203. { { 18.0, -7.0, 1.0 }, { 18, 10.0, 1.0 }, { 0, 1, 0 }, { 0, 1, 0 }, 3.0 },
204. Endpunkt Lagerfläche
205. -- ped2
206. { { -6.5, -15.0, .3 }, { -14.0, -13.0, .3 }, { 0, 1, .0 }, { -1, 1, .0 }, 3.0 },
207. { { -14.0, -13.0, .3 }, { -18, -13.0, 1.0 }, { -1, 0, 1 }, { -1, 0, 1 }, 3.0 },
208. { { -18, -13.0, 1.0 }, { -18.0, -7.0, 1.0 }, { 0, 1, 0 }, { 0, 1, 0 }, 3.0 },
209. { { -18.0, -7.0, 1.0 }, { -18, 10.0, 1.0 }, { 0, 1, 0 }, { 0, 1, 0 }, 3.0 },
210. -- connection /Hierbei handelt es sich um eine Verbindung der beiden Lagerflächen.
213. Damit wenn die Linien einmal die Seiten Tauschen die Ware auf die richtige Seite
214. wechseln kann/
215. { { -14.0, -13.0, .3 }, { 14.0, -13.0, .3 }, { 1, 0, 0 }, { 1, 0, 0 }, 3.0 },
216. ))
219. }
220. },
221. terminals = { /vehicleNodes sind die Haltepunkte der Fahrzeuge und personNodes sind die
    Endpunkte für die Verladung/
222. { vehicleNode = 6, personNodes = { 44 }, personEdges = { 22 } },
223. { vehicleNode = 20, personNodes = { 52 }, personEdges = { 26 } },
224. }
225. },
226. cost = {
227. price = 7000
228. },
229. },
230. }
233. end

```

Display More

Nachtrag1

Die Linie auf der die Fahrzeuge halten sollen muss 100% Parallel zum Lagerplatz Liegen!

Was bei meinem Mod bedeutete das der LKW auf der rechten Seite quasi direkt nach der Einfahrt stehen

bilib.

Ich habe aber eine Möglichkeit gefunden das zu umgehen

"Beispiel"

{ { 6.0, 1.0, .0 }, { 12.0, 7.0, .0 },

{ { 12.0, 7.0, .0 }, { 12, 12.0, .0 } }

{ { 12.0, 12.0, .0 }, { 12, 20.0, .0 } }

Wir gehen jetzt davon aus das diese Linien Paralel zur Lagerfläche Liegen dann würde der LKW auf { 12.0, 7.0, .0 }

zum Stehen kommen weil { 12.0, 7.0, .0 }, { 12, 12.0, .0 } Paralel zum Lagerplatz liegt

Wenn wir aber hin gehen und eine minimale Änderung vornehmen z.B.

{ { 6.0, 1.0, .0 }, { 11.9, 7.0, .0 },

{ { 11.9, 7.0, .0 }, { 12, 12.0, .0 } }

{ { 12.0, 12.0, .0 }, { 12, 20.0, .0 } }

Läuft die Linie erst ab { 12.0, 12.0, .0 }, { 12, 20.0, .0 } paralel und der LKW würde erst auf { 12.0, 12.0, .0 }

Stoppen[/code]

Das Prinzip der Bus und Tram Stationen, warscheinlich auch der Bahnhöfe, ist fast genau so aufgebaut

So ich hoffe ich habe es halbwegs Verständlich erklärt und Wünsche euch viel Erfolg beim Modden eurer eigenen Stationen 👍

MfG Grimes

Edit:

Nach neusten Erkenntnissen muss für jeden Haltepunkt eine durchgehende Linie von der Ein- bis zur Ausfahrt gemacht werden die Kurzen Linien dienen nur zur Abkürzung um zum Haltepunkt zu gelangen oder ihn zu Verlassen

Wenn jemand neue Infos hat darf er diesen Beitrag gerne Editieren

MfG Grimes