

Поиск кратчайшего пути в лабиринте с помощью библиотеки pathfinder на языке программирования Python

Чингалаев Сергей Алексеевич

Приамурский государственный университет им. Шолом-Алейхема

Студент

Аннотация

Целью данной статьи является разработка программы для поиска кратчайшего пути прохождения лабиринта на языке Python. Для достижения этой цели была использована библиотека pathfinder и язык программирования Python. В этой статье была разработана программа поиска пути в лабиринте, с помощью которой можно находить кратчайший маршрут.

Ключевые слова: Python, программа, pathfinder, поиск пути, разработка, лабиринт

Finding a path using the pathfinder library in Python programming language

Chingalaev Sergey Alekseevich

Sholom-Aleichem Priamursky State University

Student

Abstract

The purpose of this article is to develop a program for finding a path in Python. To achieve this goal, the pathfinder library and the Python programming language were used. In this article, a path finder was developed that can automate this process.

Keywords: Python, program, pathfinder, pathfinder, development

Большинство людей в современном мире пользуются системами трекинга и навигационными системами. В этой статье будет создана программа, которая будет высчитывать наиболее краткий путь, что поможет облегчить и автоматизирует поиск маршрутов.

Цель исследования: разработка программы для поиска кратчайшего пути прохождения лабиринта, с помощью библиотеки pathfinder на языке программирования Python.

О.Е. Узких, В.В. Моторин в своей статье предлагается альтернативный подход к обучению программирования на основе языка Python [1]. С.А.Слауцкий, Д.С. Королев, В.В. Шумилин в своей статье описали возможности программы Pathfinder при проведении научно-исследовательских работ [2]. М.В. Скачко, М.Г. Есина в своей статье описали


```
for i, line in enumerate(mazef.readlines()):
    digits_line = []
    for j, c in enumerate(line):
        if c == '*' or c == 's' or c == 'e':
            digits_line.append(1)
            if c == 's':
                start = (j, i)
            elif c == 'e':
                end = (j, i)
        elif c == '#':
            digits_line.append(0)
    matrix.append(digits_line)
```

Рис.3. Считывание файла

Далее создаем с переменные, с которыми будет работать библиотека поиска пути и запускаем поиск (Рис.4)

```
grid = Grid(matrix=matrix)
start = grid.node(start[0], start[1])
end = grid.node(end[0], end[1])

finder = AStarFinder(diagonal_movement=DiagonalMovement.never)
path, runs = finder.find_path(start, end, grid)
```

Рис.4. Поиск пути

Напишем функции вывода результата в консоль (Рис 5.).

```
print('Операций:', runs, 'Длина пути:', len(path))
print(grid.grid_str(path=path, start=start, end=end))
```

Рис.5. Код вывода в консоль

Запустим программу и увидим результат выполнения поиска пути в консоль (Рис 6).

```
Операций: 22 Длина пути: 18
+-----+
|sxxxxx      |
|#####x   ##  ##|
|##  xx  ##   #|
|#####x#  #  ##  #|
|###  x##  ##### ##|
|#####x## #   ##|
|##xxx#     ##### #|
|#xx#     ##      |
|#e#####|
+-----+
```

Рис.6. Результат выполнения программы

Таким образом, была написана программа, которая осуществляет поиск кратчайшего пути прохождения лабиринта, с помощью библиотеки pathfinder на языке программирования Python

Библиографический список

1. Узких О.Е., Моторин В.В. Python в качестве альтернативного языка обучения программированию // Конструктивное обучение в образовательной системе: проблемы и решения. Материалы Международной научно-практической конференции 2016. С. 150-155.
2. Слацкий С.А., Королев Д.С., Шумилин В.В. Возможности программы "pathfinder" при проведении научно-исследовательских работ // Проблемы обеспечения безопасности при ликвидации последствий чрезвычайных ситуаций. 2015. Т. 2. № 1 (4). С. 99-103..
3. Скачко М.В., Есина М.Г. Описание программы pathfinder, используемой для обеспечения техносферной безопасности // Сборник материалов III Всероссийской научно-практической конференции с международным участием. 2018. С. 258-260.
4. Miller J.A. Promoting computer literacy through programming python. University of Michigan, 2004