

AAE1001 - Introduction to Artificial Intelligence and Data Analytics in Aerospace and Aviation Engineering

Week 8 – Software Installation and Setup (Windows)

Dr. Guohao Zhang, Assisted by

Dr Penghui XU, Mr Zekun ZHANG, Mr Di HAI, Mr Yidi CHEN, Mr Zhen LYU

Software Installation and setup Guide

Install VS code in Windows 10/11

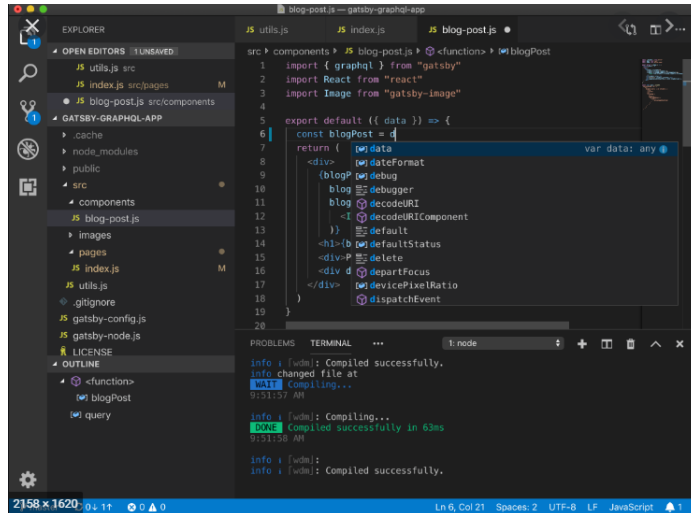
Step 1: Download the latest VS code

<https://code.visualstudio.com/download>

Tutorial Video:

https://www.youtube.com/watch?v=MZ2w7VU9T4c&ab_channel=POLYUIPNL

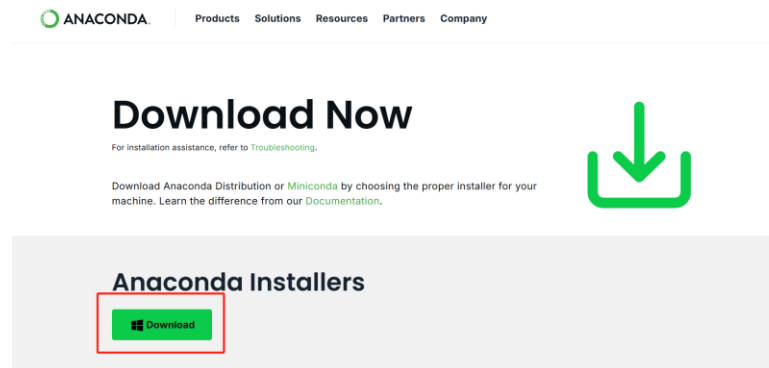
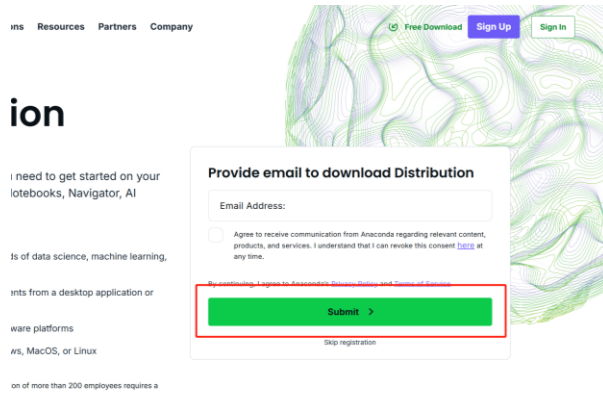
Step 2: Install latest VS code in Windows 10/11



Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

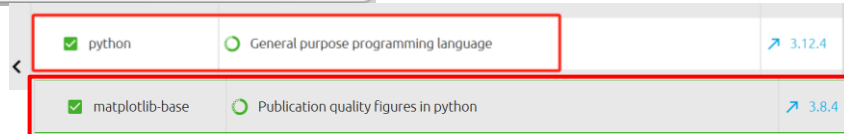
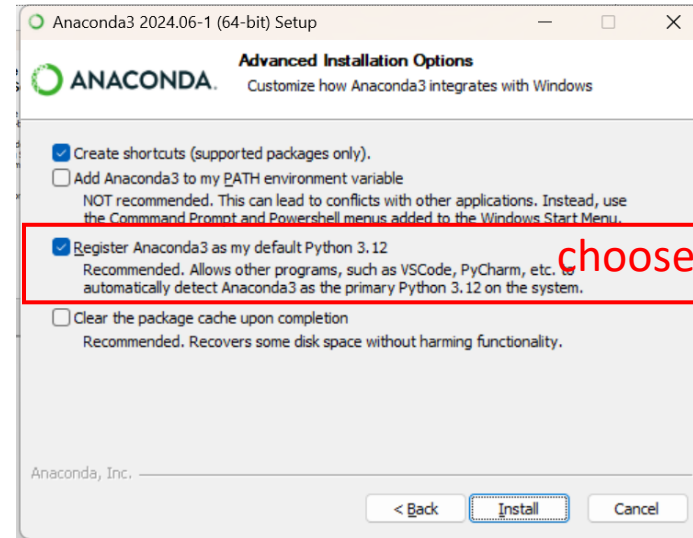
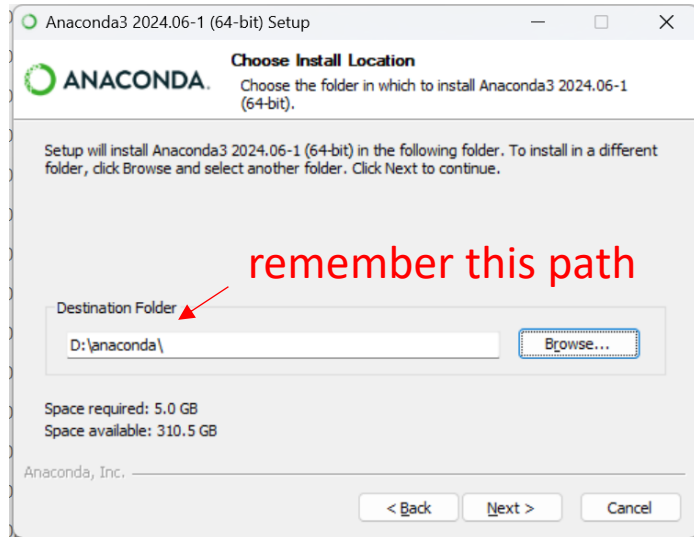
Install Anaconda for Python management

- <https://www.anaconda.com/download>
- **Anaconda** is an open source Python and R distribution designed for scientific computing, data science, machine learning, and big data analysis. It provides a complete set of tools and libraries to facilitate data processing and scientific computing. **It can manage different Python versions.**

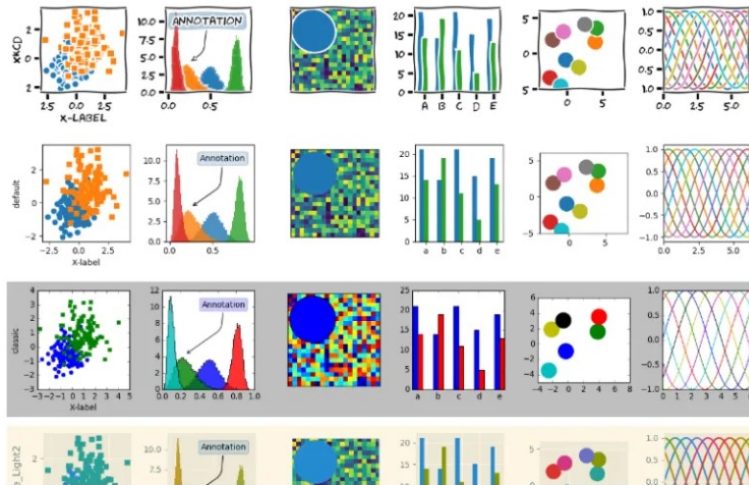


Install Anaconda for Python management

- The process of **Anaconda** installation can automatically install the newest **Python** and **Matplotlib**

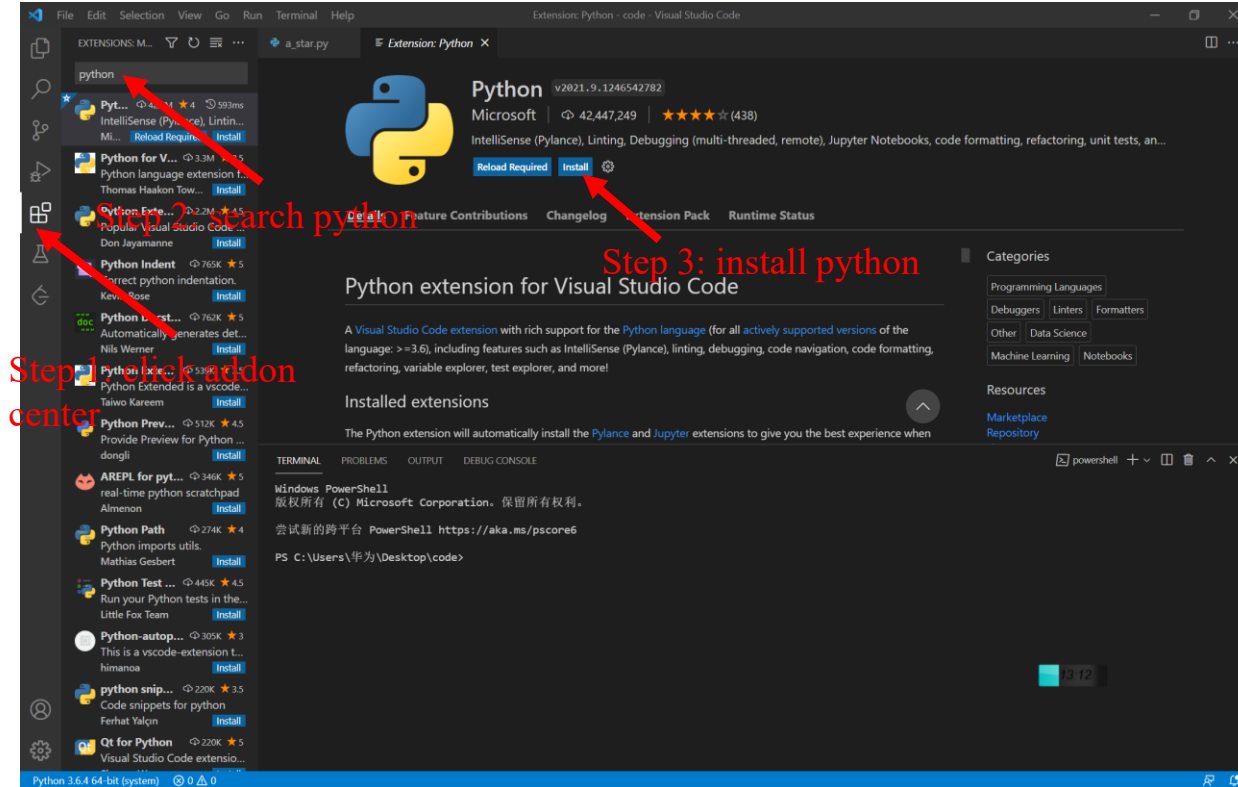


Matplotlib introduction



Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+.

Install Python **addon** in VS code in Windows 10/11



VS code setting

The image shows a VS Code interface with the Extensions view on the left. The 'Python' and 'Python Debugger' extensions are installed. The command palette is open, showing the command '>Python: Select Interpreter'. The right panel shows the 'Select Interpreter' dialog with 'Python 3.12.4 (base)' selected. The path 'D:\anaconda\python.exe' is also visible in the dialog.

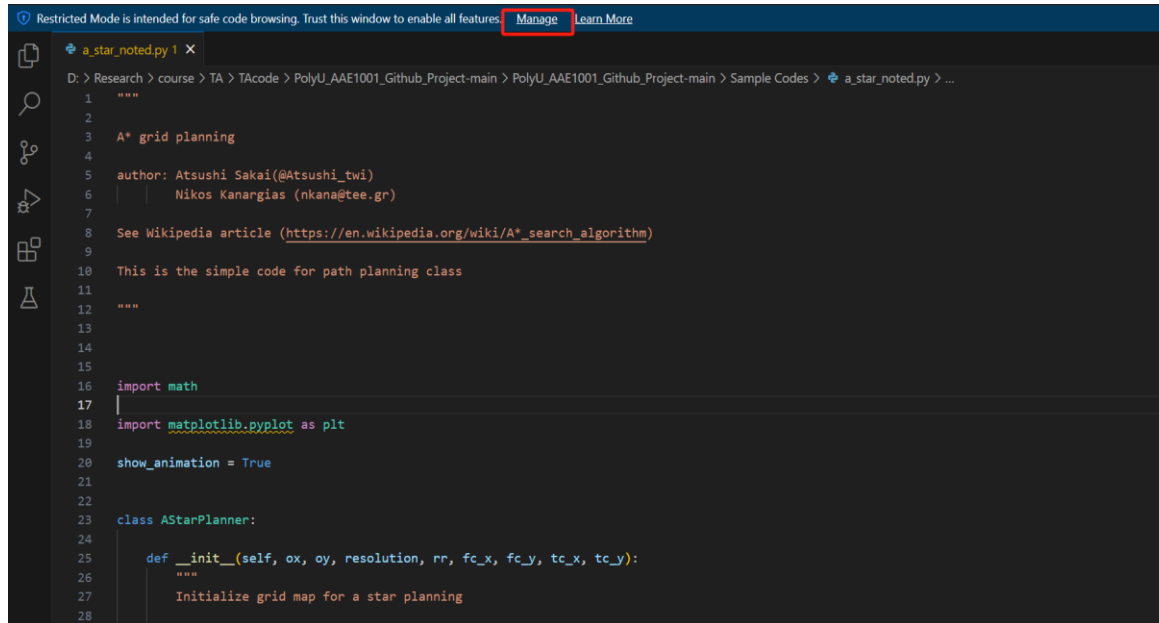
1. Type: ">Python: Select Interpreter"

2. Select "Python 3.12.4" installed by anaconda

Run A Star in VS code

a_star_noted.py

➤ step 1: Open the code sample by VS code



```
1  """
2
3  A* grid planning
4
5  author: Atsushi Sakai (@Atsushi_twi)
6         Nikos Kanargias (nkana@tee.gr)
7
8  See Wikipedia article (https://en.wikipedia.org/wiki/A\*\_search\_algorithm)
9
10 This is the simple code for path planning class
11
12 """
13
14
15
16 import math
17
18 import matplotlib.pyplot as plt
19
20 show_animation = True
21
22
23 class AStarPlanner:
24
25     def __init__(self, ox, oy, resolution, rr, fc_x, fc_y, tc_x, tc_y):
26         """
27         Initialize grid map for a star planning
28
```

In a Trusted Window

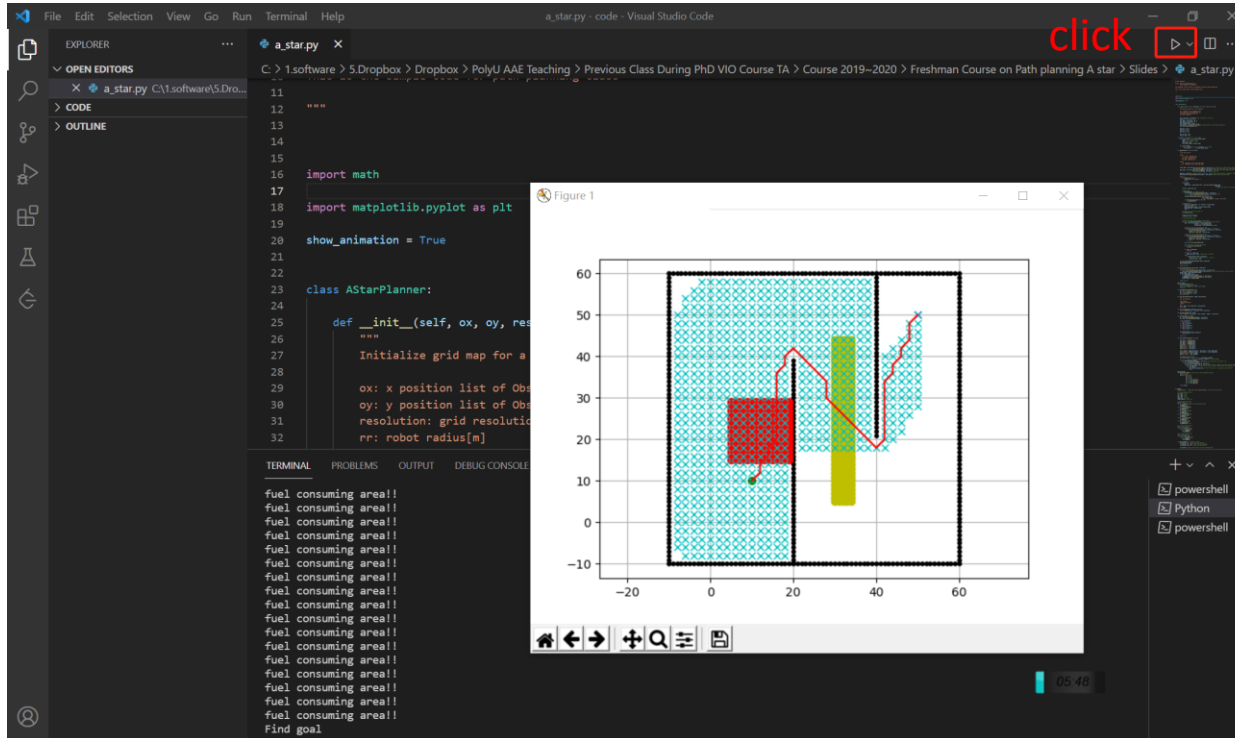
You trust the authors of the files in the current window. All features are enabled:

- ✓ Tasks are allowed to run
- ✓ Debugging is enabled
- ✓ All enabled extensions are activated

Trust

Ctrl+Enter

➤step 2: Run the demo



Run A Star in VS code

Run the code

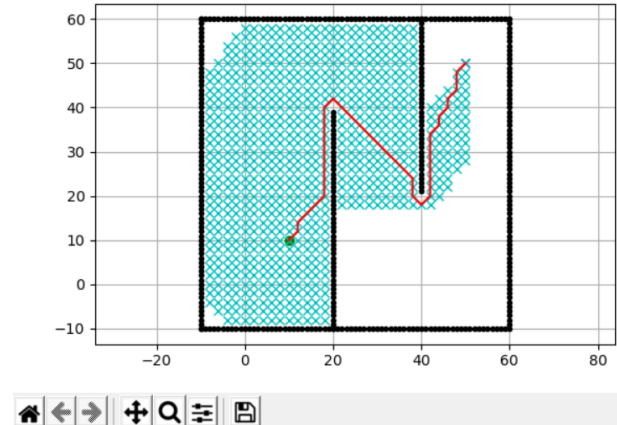
```

a_star.py - PythonRobotics-master - Visual Studio Code
a_star.py x
PathPlanning ? AStar > a_star.py > AStarRunner > planning
self.calc_obstacle_map(ox, oy)
18
19
20
21 class Node:
22     def __init__(self, x, y, cost, parent_index):
23         self.x = x # index of grid
24         self.y = y # index of grid
25         self.cost = cost
26         self.parent_index = parent_index
27
28     def __str__(self):
29         return str(self.x) + "," + str(self.y) + "," + str(
30             self.cost) + "," + str(self.parent_index)
31
32 def planning(self, sx, sy, gx, gy):
33     """
34     A star path search
35
36     input:
37     sx: start x position [m]
38     sy: start y position [m]
39     gx: goal x position [m]
40     gy: goal y position [m]
41
42     output:
43     rx: x position list of the final path
44     ry: y position list of the final path
45     """
46
47     start_node = self.Node(self.calc_xy_index(sx, self.min_x),
48                             self.calc_xy_index(sy, self.min_y), 0.0, -1)
49     goal_node = self.Node(self.calc_xy_index(gx, self.min_x),
50                             self.calc_xy_index(gy, self.min_y), 0.0, -1)
51
52     open_set, closed_set = dict(), dict()
53     open_set[self.calc_xy_index(start_node.x, start_node.y)] = start_node

```

A* is a graph traversal and path search algorithm, which is often used in many fields of computer science due to its completeness, optimality, and optimal efficiency. One major practical drawback is its space complexity, as it stores all generated nodes in memory.

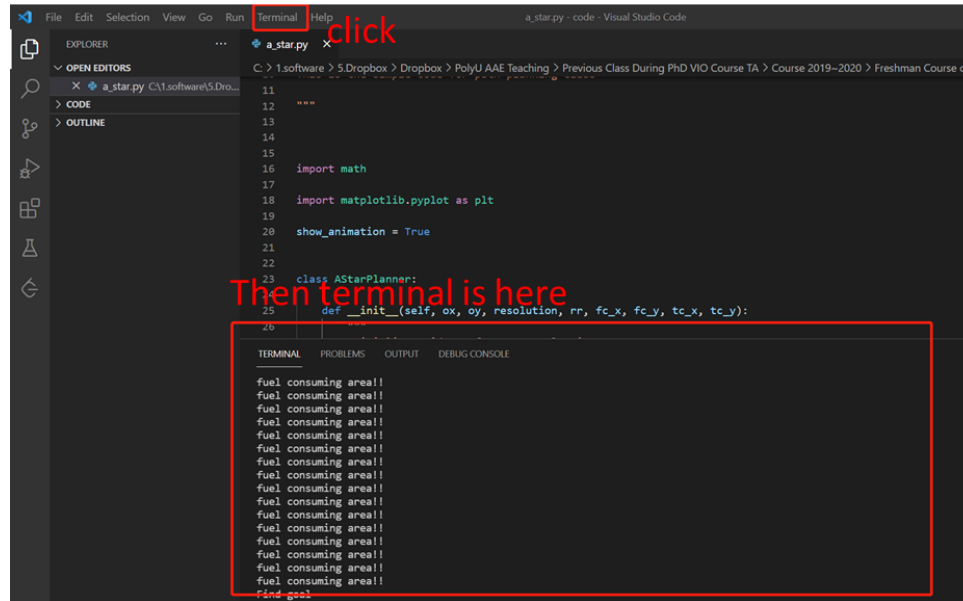
Figure 1



Troubleshoot: Connect Github page via command in VS Code

➤ Input the command below to the terminal (change the blue to your info)

- *git config --global user.name* *aaestudent*
- *git config --global user.email* *aaestudent@gmail.com*



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a file named `a_star.py`. The main editor area displays the code for `a_star.py`, which includes imports for `math` and `matplotlib.pyplot`, and a class `AStarPlanner`. A red box highlights the `terminal` tab in the top bar, with a red arrow pointing to it and the text "click". Another red box highlights the terminal window at the bottom, which contains the output of a command: `fuel consuming area!!` repeated multiple times. A red arrow points to the terminal window with the text "Then terminal is here".