

HW1

jdave 76

February 2026

1 Q1 image

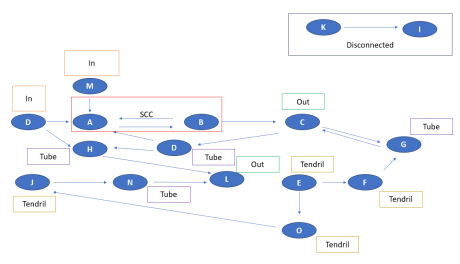


Figure 1: Q1.

Q1 questions

SCC: A, B

IN: D, M

OUT: C, L

Tendrils:

E (can reach out), F(reachable from IN/can reach OUT), J(reachable from IN/can reach OUT), O (reachable from IN/can reach OUT)

Tubes:

D(connect C to A), G(connects F to C), H(connects D to L)

Disconnected: I, K

2 Q2



Figure 2: Q2a.

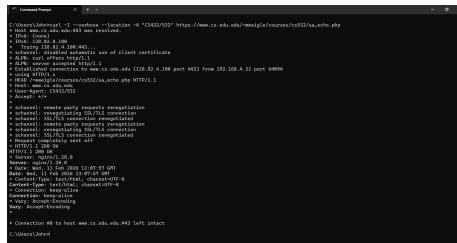


Figure 3: Q2b.



Figure 4: Q2c.

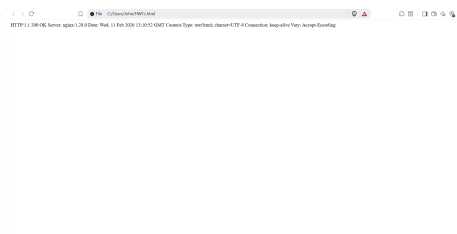


Figure 5: Q2d.

3 Q3

Listing 1: Generate 500 Unique URIs

```

1 from bs4 import BeautifulSoup
2 import requests
3 from urllib.parse import urljoin
4 import random
5
6 url = "https://norfolk.gov" # Replace with target URL
7 base_url = url
8
9 response = requests.get(url)
10 soup = BeautifulSoup(response.content, 'html.parser')
11
12 target_unique = 500
13 unique_uris = set()
14 uri_list = []
15
16 # Extract all links
17 links = soup.find_all('a', href=True)
18 uri_list = []
19
20
21 while len(uri_list) < 500:
22     for link in links:
23         try:
24             head_response = requests.head(url, allow_redirects=True
25                                     , timeout=20)
26             content_type = head_response.headers.get('Content-Type'
27                                     , '')
28             content_length = int(head_response.headers.get('Content
29                                     -Length', 0))
30
31             # Check if it's HTML (case-insensitive)
32             if 'text/html' in content_type.lower() and
33                 content_length > 1000:
34                 uri_list.append(link)
35                 seed_value = hash(tuple(uri_list)) # Convert list
36                                     to tuple (hashable) and hash it
37                 random.seed(seed_value)
38         except FileNotFoundError:
39             print("File not found.")

```

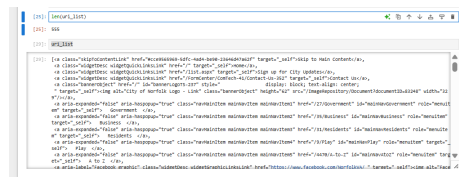


Figure 6: over 500 Unique URIs.

I had my program randomly select URIs as the new random seed.