# CSE 419L – Information Retrieval lab Week-5

**G.Harsha Vardhan AP20110010689 CSE J**

Q2. Implement the above question using skip lists.

**Dataset**

Textbook : Hadoop the definitive guide

The text book was splitted into 100 text documents

**Code**

#5b

import os import re

class SkipListNode:

def init (self, doc\_id=None): self.doc\_id = doc\_id self.next = None

class InvertedIndex: def init (self):

self.index = {}

def add\_document(self, doc\_id, text):

tokens = re.findall(r'\w+', text.lower()) for token in set(tokens):

if token not in self.index: self.index[token] = SkipListNode()

current = self.index[token] while current.next:

current = current.next current.next = SkipListNode(doc\_id)

def index\_directory(self, directory\_path):

for root, \_, files in os.walk(directory\_path): for file in files:

file\_name = os.path.splitext(file)[0] file\_path = os.path.join(root, file)

with open(file\_path, 'r', encoding='utf-8') as f: text = f.read()

self.add\_document(file\_name, text)

def search(self, query): query = query.lower() if query in self.index:

current = self.index[query].next results = []

while current: results.append(current.doc\_id) current = current.next

return results else:

return []

def display\_index(self):

for token, node in self.index.items(): print(f"Word: {token}")

current = node.next print("Doc ID: ",end="") while current:

print(f"{current.doc\_id},",end="") current = current.next

print()

index = InvertedIndex() index.index\_directory('/content/drive/MyDrive/ir\_dataset')

# Display the inverted index index.display\_index()

**Output**

