

EVOLUTION IN PARALLEL

- 1.K.Devi Madhumitha (21B01A0585) : CSE-B
- 2.R.Nikhila (21B01A05F1) : CSE-C
- 3.Koram Siri (21B01A1286) : IT-B
- 4.K.S.V.Sahithi (21B01A0586) : CSE-B
- 5.R.Pavani Durga (21B01A05F2) : CSE-C

SVECW

10-02-2023

Introduction

- We have to contradict the hypothesis and prove that there can be **Two Evolutionary Paths** in which a **New Species** is formed.

Approach

- We Arranged the input Genetic codes in the Order of their length.
- We created first method to check whether the Genetic code fits to other or not.
- We define diff() method to seperate the paths of evolution.

Learnings

- In *PYTHON* our code gets little easier when compared to other programming languages.
- In our code we used a package called **SYS**.
- we learned how to work on **Gitlab** .
- we also learned to prepare presentation using **LaTex**.

Challenges

- Firstly we found a challenge for understanding the problem statement.
- Then the next challenge was to build a code which gives the desired output.
- Optimisation of code was another challenge we had to encounter.

Statistics

- Number of Lines of Code : 65
- Number of Functions : 3

And the functions are :

- *is evolved()*
- *diff()*
- *store()*

Demo/ Screen Shots

- Demo of the Project

Command Prompt

```
Microsoft Windows [Version 10.0.22000.1455]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\91812>notepad evolution_in_parallel.py  
  
C:\Users\91812>python3 evolution_in_parallel.py 5 AACCMMAA ACA MM ACMMAA AA A  
(4, 1, ['A', 'AA', 'ACA', 'ACMAA'], ['MM'])  
  
C:\Users\91812>
```

● Screen Shots of the Project

p1 - Notepad

File Edit Format View Help

```
import sys
def is_evolved(sample, current_species):
    _ = __ = 0
    while _ < len(sample):
        if __ == len(current_species):
            return False
        if sample[_] == current_species[__]:
            _ += 1
            __ += 1
    return True
def diff(fossils_found):
    lst1 = []
    lst2 = []
    for _ in range(1, MAX):
        if fossils_found[_][0] == 0:
            continue
        else:
            if fossils_found[_][1] == 0:
                if len(lst1) == 0:
                    lst1.append(fossils_found[_][0])
                elif len(lst2) == 0:
                    if is_evolved(lst1[-1], fossils_found[_][0]):
                        lst1.append(fossils_found[_][0])
                    else:
                        lst2.append(fossils_found[_][0])
            else:
                if not is_evolved(lst1[-1], fossils_found[_][0]):
                    lst2.append(fossils_found[_][0])
                elif not is_evolved(lst2[-1], fossils_found[_][0]):
                    lst1.append(fossils_found[_][0])
                else:
                    return -1
```



```

else:
    if len(lst1) == 0:
        lst1.append(fossils_found[_][0])
        lst2.append(fossils_found[_][1])
    elif len(lst2) == 0:
        if is_evolved(lst1[-1], fossils_found[_][0]):
            lst1.append(fossils_found[_][0])
            lst2.append(fossils_found[_][1])
        elif is_evolved(lst1[-1], fossils_found[_][1]):
            lst1.append(fossils_found[_][1])
            lst2.append(fossils_found[_][0])
        else:
            return -1
    return len(lst1),len(lst2),lst1,lst2

def store():
    result = True
    for i in range(3,no_of_fossils_found+3):
        fossil_sample = sys.argv[i]
        fs = len(fossil_sample)
        if not is_evolved(fossil_sample, current_species) or (fossils_found[fs][0] and fossils_found[fs][1]):
            result = False
            return -1
        elif fossils_found[fs][0] == 0:
            fossils_found[fs][0] = fossil_sample
        elif fossils_found[fs][1] == 0:
            fossils_found[fs][1] = fossil_sample
    if result:
        return diff(fossils_found)

MAX = 4000
no_of_fossils_found = int(sys.argv[1])
current_species = sys.argv[2]
fossils_found = [[0, 0] for _ in range(0, MAX)]
print(store())

```

