Ve484 machine Problem Report

侯越超 Hou Yuechao 5133709265

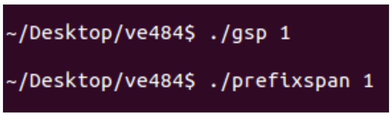
i. Demonstration

In the first test, I only use a database of one size with minsup equal to 1 to test whether my program works well or not:

ACGT

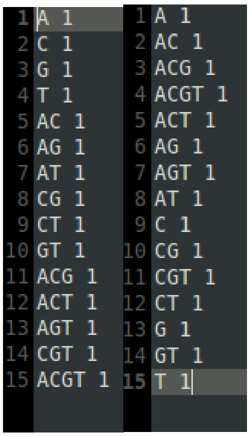
Hence, with my expectation, the result should include $2^4 - 1 = 15$ subsequences.

The result is shown in the following:



```
~/Desktop/ve484$ ./gsp 1

~/Desktop/ve484$ ./prefixspan 1
```

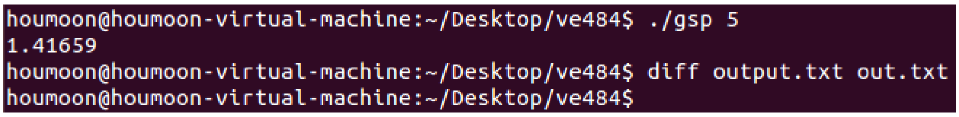


```
A 1
C 1
G 1
T 1
AC 1
AG 1
AT 1
CG 1
CT 1
GT 1
ACG 1
ACT 1
AGT 1
CGT 1
ACGT 1
```

```
A 1
AC 1
ACG 1
ACGT 1
ACT 1
AG 1
AGT 1
AT 1
C 1
CG 1
CGT 1
CT 1
G 1
GT 1
T 1
```

From the pictures above, we can see that each of the algorithms works well, and both of them list all the results we expected.

Next, I use the sample test given by the TA to test whether my gsp algorithm works well or not.



```
houmoon@houmoon-virtual-machine:~/Desktop/ve484$ ./gsp 5
1.41659
houmoon@houmoon-virtual-machine:~/Desktop/ve484$ diff output.txt out.txt
houmoon@houmoon-virtual-machine:~/Desktop/ve484$
```

Form the picture above, it shows that the result of my gsp algorithm is the same with the answer given by TAs.

ii. Testing

For the second to the fourth testing cases, I use the duplicates of the sample test given by TA. For the last testing case, I just used the sample test given by the Professor on Sakai.

Dataset size	minsup	Running Time (s)	
		GSP	PrefixSpan
3	3	1.75094	0.720762
	2	20.4969	5.61112
30	30	0.008607	0.0068
	15	1.0289	0.624982
60	60	0.015491	0.009266
	30	2.26121	1.54522

90	90	0.011764	0.009116
	45	2.59122	0.990521
1062	1062	0.67769	1.181488
	1059	11.4673	17.8778

iii. Conclusion

From the testing result above, we can see that when the size of the dataset is small, the running time of the gsp is faster than that of the prefixspan. However, when the size of dataset is quite large the performance of the gsp is better than that of the prefixspan.

(The reason may be that I used unordered map to do the finding job in the gsp programming, whose performance is well when the size of the database is large.)

iv. README

This program is a demo for gsp algorithm and prefixspan algorithm, written based on C++ 11 in the environment of Ubuntu 14.04.

The Makefile is included, so each time when you want to run the program, first type:

```
make
```

in the terminal under the same directory of the program files to compile the programming files.

The makefile will produce two .o files gsp and prefixspan standing for gsp algorithm and prefixspan algorithm respectively. To run the program, type:

```
./gsp <minsup>
./prefixspan <minsup>
```

in the terminal under the same directory, where <minsup> is replaced by the minsup designed by the user. Then the program will read the data from the file named "sequencedb.txt" and output the result to the file named "output.txt" and output the running time to the terminal in the unit of second.