Co 225 – Lab 03 – Part C

E/18/180 – Kodituwakku M.K.N.M.

• Lab03PartC.java was implemented by using Map

• Lab03PartD.java was implemented by using List

• Lab03PartE.java was implemented by using Set

**Running times**

Part C-Map

When inputting a E-number

**Text

Description automatically generated**

When inputting a letter

Text

Description automatically generated

* A Map is an object that maps keys to values. A map cannot contain duplicate keys: Each key can map to at most one value. It models the mathematical function abstraction.

Part D-List

When inputting a E-number

Text

Description automatically generated

When inputting a letter

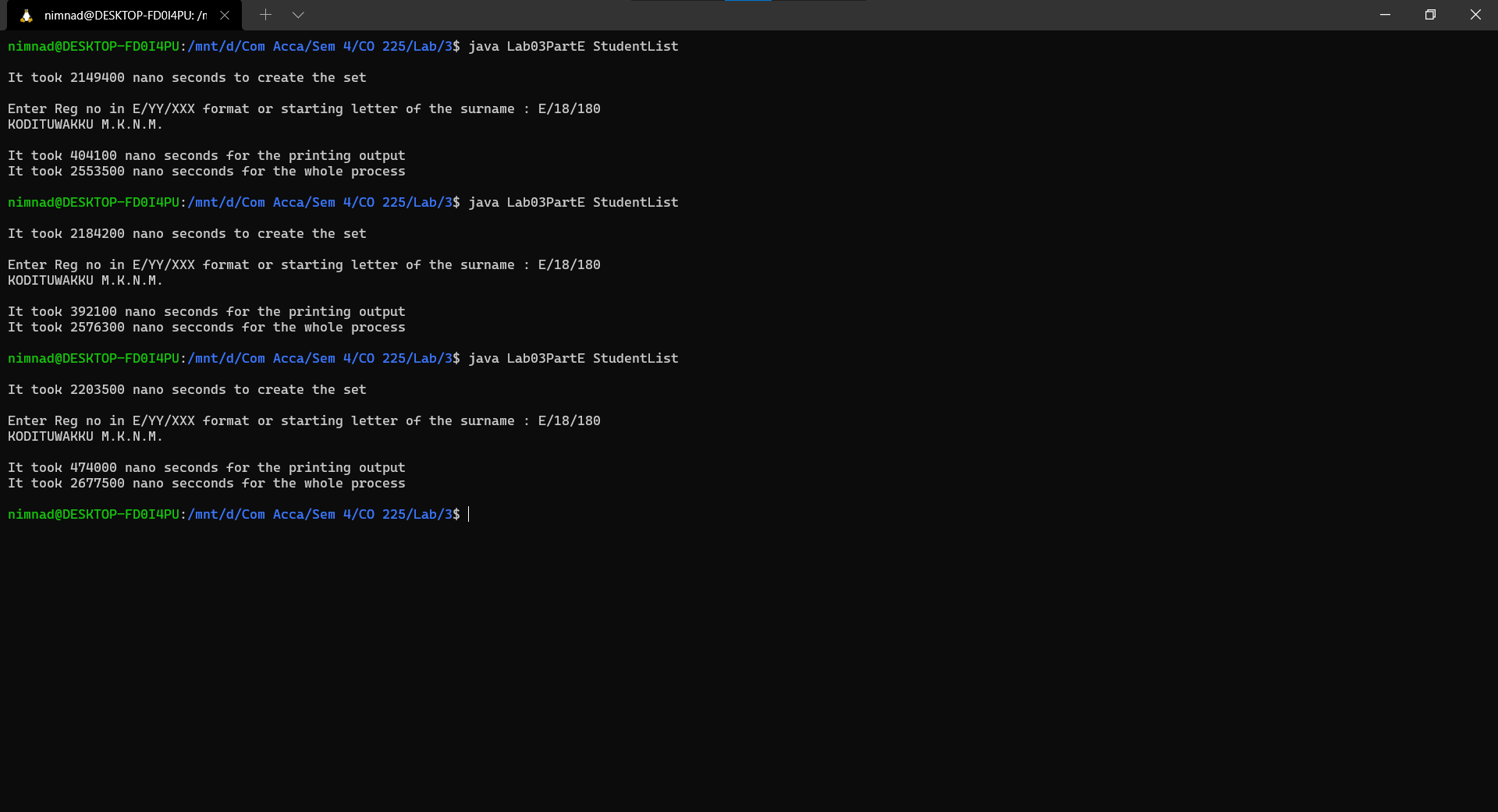
Text

Description automatically generated

* List in Java provides the facility to maintain the ordered collection. It contains the index-based methods to insert, update, delete and search the elements. It can have the duplicate elements also.

Part E-Set

When inputting a E-number



When inputting a letter

Text

Description automatically generated

* The set is an interface available in the java.util package. The set interface extends the Collection interface. An unordered collection or list in which duplicates are not allowed is referred to as a collection interface. The set interface is used to create the mathematical set. The set interface use collection interface's methods to avoid the insertion of the same elements.

Conclusion

Calculating average times using above results.

1. Time to insert data into different data instructions

|  |  |  |  |
| --- | --- | --- | --- |
| **Trial** | **Making time in nanoseconds** | | |
| **Map** | **List** | **Set** |
| 1 | 1896100 | 2101700 | 2149400 |
| 2 | 2072400 | 1870600 | 2184200 |
| 3 | 1750500 | 2219200 | 2203500 |
| 4 | 1865700 | 1998000 | 2104400 |
| 5 | 1956700 | 2226200 | 2037600 |
| 6 | 1817400 | 2250700 | 2063700 |
| **Avg. Time** | **1893133.333** | **2111066.667** | **2123800** |

* From above results inserting data into map is faster than other two.

1. Time taken by two operations in different data structures

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Trial** | Map | | List | | Set | |
| **Enum as input** | **Letter as input** | **Enum as input** | **Letter as input** | **Enum as input** | **Letter as input** |
| 1 | 383100 | 22294600 | 291900 | 18633400 | 404100 | 17702800 |
| 2 | 330700 | 19008900 | 397500 | 19923900 | 392100 | 18579600 |
| 3 | 346300 | 22453300 | 351500 | 19527300 | 474000 | 19720000 |
| **Avg. Time** | **353366.6667** | **21252266.67** | **346966.6667** | **19361533.33** | **423400** | **18667466.67** |
| **Overall Avg. Time** | 10802816.67 | | 9854250 | | 9545433.333 | |

From the results we can obtain that functionality is fast is Set implementation.