**Problem Statement:**

1. Write a prolog program to implement Bubble Sort.
2. Write a prolog program to implement Insertion Sort.
3. Write a prolog program to implement Quick Sort.

**Design/Methodology/Algorithm/Data Structure**

1. **Domains**

list = integer\*.

Predicates

bubblesort(list,list).

swap(list,list).

printlist(list).

**Clauses**

bubblesort(InputList,SortList) :-

swap(InputList,List) , ! ,

printlist(List),

bubblesort(List,SortList).

bubblesort(SortList,SortList).

swap([X,Y|List],[Y,X|List]) :- X > Y.

swap([Z|List],[Z|List1]) :- swap(List,List1).

printlist([]) :- nl.

printlist([Head|List]) :-

write(Head, \" \"),

printlist(List).

**Domains**

list = integer\*.

element = integer.

**Predicates**

insertSort(list, list).

insertItem(element, list, list).

printlist(list).

**Clauses**

insertSort([H|List], Result) :-

insertSort(List, Temp),

printlist(Temp),

insertItem(H, Temp, Result).

insertSort([], []).

insertItem(X, [H|List], [H|Result]) :-

H < X, !,

insertItem(X, List, Result).

insertItem(X, List, [X|List]).

printlist([]) :- nl.

printlist([X|List]) :-

write(X, \" \"),

printlist(List).



**Domains**

list = integer\*.

**Predicates**

quicksort(list,list).

split(integer,list,list,list).

concatenate(list,list,list).

printlist(list).

**Clauses**

quicksort([],[]).

quicksort([Head|Tail],SortedList) :-

split(Head,Tail,SList,BList),

quicksort(SList,SList1),

quicksort(BList,BList1),

concatenate(SList1,[Head|Blist1],SortedList),

printlist(SortedList).

split(\_,[],[],[]).

split(Item,[Head1|Tail1],[Head1|SList],BList) :-

Item > Head1 , ! ,

split(Item,Tail1,SList,BList).

split(Item,[Head1|Tail1],SList,[Head1|BList]) :-

split(Item,Tail1,SList,BList).

concatenate([],List,List).

concatenate([Item|List1],List2,[Item|List3]) :-

concatenate(List1,List2,List3).

printlist([]) :- nl.

printlist([Head|Tail]) :-

write(Head,\" \"),

printlist(Tail).

**Output :**

1. Goal: BubbleSort([2,4,1,3,5,9,6],L).

2 1 4 3 5 9 6

1 2 4 3 5 9 6

1 2 3 4 5 9 6

1 2 3 4 5 6 9

L = [1,2,3,4,5,6,9]

1 Solution

1. Goal: insertsort([2,4,1,3,5,9,6],L).

2 4 1 3 5 9 6

1 2 4 3 5 9 6

1 2 3 4 5 9 6

1 2 3 4 5 6 9

L = [1,2,3,4,5,6,9]

1 Solution

1. Goal: quicksort([2,4,1,3,5,9,6],L).

1

3

6

6 9

5 6 9

3 4 5 6 9

1 2 3 4 5 6 9

L=[1,2,3,4,5,6,9