Assignment no	1
Aim	**In second year computer engineering class, group A student's play cricket, group B students play badminton and group C students play football.** **Write a Python program using functions to compute following: - a) List of students who play both cricket and badminton b) List of students who play either cricket or badminton but not both c) Number of students who play neither cricket nor badminton d) Number of students who play cricket and football but not badminton. (Note- While realizing the group, duplicate entries should be avoided, Do not use SET built-in functions)**
Objective	To understand the concept of functions in programming languages To understand , implement SET data structure and its operations To use list or array in python to implement derived SET data structures
Outcome	To understand ,design and implement SET data structure using list or array in python To write/implement user defined functions/modules for different operations of SET in python To write menu driven, modular program in Python
OS/Programming tools used	(64-Bit) 64-BIT Fedora 17 or latest 64-BIT Update of Equivalent Open source OS or latest 64-BIT Version and update of Microsoft Windows 7 Operating System onwards Programming Tools (64-Bit) Eclipse with Python plugin or Pycharm IDE

Theory related to assignment:

In this assignment we will implements derived data structure SET data structure using list or array as primitive data structure (NOTE: we are not supposed to use inbuilt SET in python)

A Set is an unordered collection data type that is iterable, mutable and has no duplicate elements.

Lists

- Python lists are very flexible and can hold arbitrary data.
- Lists are a part of Python’s syntax, so they do not need to be declared first.
- Resize quickly
- Store heterogeneous data
- Mathematical functions can be applied directly
- List consume more memory

Arrays

- Arrays need to first be imported, or declared, from other libraries (i.e. numpy).
- Store homogenous data
- Wide range of mathematical functions can be applied directly
- Arrays are compact in size

How to use list in python:

```
lst=[ ]// empty list
lst.append(1) //append or add 1 element to list
print(lst) //print the list
```

How to use array in python:

```
import array as arr
a=arr.array('i',[1,2,3]) // a is array of integer
```

We can use list as array but we can restricts to type of elements

ADT :

Set is Objects or value definition: A finite collection with zero or more elements
Functions or operator definition:
For all S є Set and item є element
createSet() : Set //creates the empty Set
addEle(Set,item) :void //adds unique items to Set
Union(Set,Set):Set // Returns union of 2 sets
isemptySet(Set):Boolean //if Set is empty returns TRUE else returns FALSE
Intersection(Set): Set //Returns Intersection of 2 sets
End Set

ADT representation using class for Set

```
Class SET {
int *a; // value definition in ADT
int n; //indicate size of set
public: //Operator definition in ADT
Set create(); //creates the empty Set with n=0 and allocate memory for array a
void addelement( i,val); // appends val in Set at ith location provided val is not present in set
and increment n;
void addelement(val) // append val at end and increment n
void read(); //read whole set
int read(i); // returns value at ith location from Set
bool exists(item); // if item exists in Set returns true else false
Set union(Set ,Set) // return union of 2 sets
Set intersection(Set ,Set) // return intersection of 2 sets
}
```

Step for implementation:

Step 1:

Program should defined six list variables which are empty initially. Declare the List for 1) students who play both cricket 2) students who play badminton 3) students who play football 3) students who play both cricket and badminton 4) students who play either cricket or badminton but not both 5) Number of students who play neither cricket nor badminton 6) Number of students who play cricket and football but not badminton

Step 2:

Accept the three sets for cricket, badminton and football by calling createSet() function which will take care of unique elements

Step 3: Define functions of union, intersection, difference/subset which accept the two list as arguments and return the answer list

Step 4: call corresponding functions with list as arguments according to menu.

Pseudo Code:

Union of two set:

procedure Set union(Set s1, Set s2)

Purpose : Finds the union of sets s1 and s2 represented as objects of Set with length of set as ‘n’

Pre-condition: nothing

Post condition: Union of two sets s3 is returned as Set

```
begin
for i=0 to i=s1.n-1
begin
    s3.addelement(s1.read(i))
end for
for i=0 to i=s2.n-1
begin
    if(s3.exists(s2.read(i))==false)
        s3.addelement(s2.read(i))
    end if
end for
return s3
```

end union

Intersection of two sets:

Set intersection(Set s1, Set s2)

Purpose : Finds the union of sets s1 and s2 represented as objects of Set with length of set as ‘n’

Pre-condition: nothing

Post condition: intersection of two sets s3 is returned as Set

```
begin
  s1.read();
  s2.read();
  for i=0 to i=s1.n -1
  begin
    if(s2.exists(s1.read(i))==true) s3.addelement(s1.read(i))
    end if
  end for
  return s3
```

end intersection

Subset of two sets:

Procedure Boolean subset (Set s1, Set s2)

Purpose : Finds the whether set2 is subset of of sets s1

Pre condition: nothing

Post condition: if set s2 is subset of set s1 then it is returned as true

```
begin
  Subsetflag=True
  for i=0 to s2.length-1 do
    If s1.exists(s2[i])==0
      Subsetflag=false
  end for
  return Subsetflag;
```

end subset;

Time Complexity: O(n)

Space Complexity: O(n)

Conclusion:

The functions for different set operations are implemented successfully using list as primitive data structure.

Review Questions:

1. What is set data structure and its applications?
2. Explain the different operations of set data structure?
3. How to use list to implement SET data structure?
4. How to use array to implement SET data structure?
5. What is list data type in python?

6. Can we add duplicate elements in SET?
7. Can we add duplicate elements in list or array?
8. SET is homogeneous or heterogeneous data type?
9. Explain the complexities of different operations of SET?