CS513 DS LAB

LAB 1 (6/AUG/22)

Max Marks: 115points

(5points :Indentation, 5points:Documentation, 5m:Good coding practice)

Time : 3 hours

Q1.) Given a input file containing the characters '(', ')', '{', '}' , '[' , ']' , and several other, read the input file **(input.txt**) and determine if the input file is having valid Parentheses or not**. (40 points)**

(Note: read file name using command line argument)

An input file is valid if:

Open brackets must be closed by the same type of brackets.

Open brackets must be closed in the correct order.

Example 1:

Input:	Output: Input file is valid Explaination: As the input file contains parantheses in correct order and opening and closing of brackets are same :<> ( ){ {} }

```
#include<stdio.h>
Int main( )
{
   int a[2];
  for(int i=0;i<2;i++)
  {
     a[i] = i;
  }
   return 0;
}
```

Example 2:

Input file:	Output: Input file is invalid Explaination: It is invalid because at line no. 4 ‘)’ do not have the same closing bracket and at line 20 do not have same closing bracket for ‘{ ‘ at line 9.

```
class S
 {
    public:
          int reverse(int x   // ‘)’ is missing
         {
            int y=0;
            int a[10];
            while(x)
           {
              if(y>INT_MAX/10 || y<INT_MIN/10)
              {
                 return 0;
              }
              else
             {
                 y=y*10 +x%10;
                 x=x/10;
              }
           )

            return y;
        }
};
```

Q2.)Implement set using **link list**, you have to perform all the below task through function call. The tasks are –**(60 points)**

1.**int makeSet(struct Set *SetA)**, [To make a head of the link list]

2. **int insert (struct Set *SetA, int value)**, [To insert value 'd' in set 'S' , if not exist] **(8 points)**

3. **int delete (struct Set *SetA, int value)**, [To delete value 'd' from set 'S' if it exists] **(8 points)**

4. **int printSet (struct Set *SetA)**, [To print the values of set] **(6 points)**

5. **int Union (struct Set *SetA, struct Set *SetB, struct Set *SetC)**, [To perform union between two sets S1, S2 and return another set]**(10 points)**

6. **int intersection (struct Set *SetA, struct Set *SetB, struct Set *SetC)**, [To perform intersection between two sets S1, S2 and return another set] (**10 points)**

7. **int difference(struct Set *SetA, struct Set *SetB, struct Set *SetC)**, [to perform set difference S1 - S2 and] (**10 points**)

8. **int deleteSet (struct Set *SetA)**, [To perform delete a set] **(8 points)**

Note:- <u>Return 1 for successfully call the function, otherwise 0.</u>

Partial code:-

```
#include<stdio.h>
struct Set
{
  int value;
  struct Set* next;
};
int makeSet(struct Set *SetA){
//Write your code
}
int insert (struct Set *SetA, int value){ // insert value in the SetA if it is not exist and return 1, otherwise return 0
//Write your code
}
int delete (struct Set *SetA, int value){ // delete value from SetA if it is exist and return 1, otherwise return 0,
//Write your code
```

}

int printSet (struct Set *SetA){ //Print all the element of the SetA, where SetA is headpointer of link list

//Write your code

}

int Union (struct Set *SetA, struct Set *SetB, struct Set *SetC){ // To perform union between two sets SetA, SatB and //store the result in SetC, , where SetA,SetB,SetC is the head pointer of the link list

//Write your code

}

int intersection (struct Set *SetA, struct Set *SetB){ // To perform intersection between two sets SetA, SatB and //store the result in SetC, , where SetA,SetB,SetC is the head pointer of the link list

//Write your code

}

int difference(struct Set *SetB, struct Set *SetB// To perform difference between two sets SetA, SatB and store the //result in SetC, where SetA,SetB,SetC is the head pointer of the link list

//Write your code

}

int deleteSet(struct Set *SetA){ //Take headpointer of link list SetA and delete all the node from heap.

//Write your code

}

int main(){

// Read text file

//Call accordingly each function

//For each line, first number represent the operation, and remaining are parameter

}