Compiler Design Lab
Lab 2

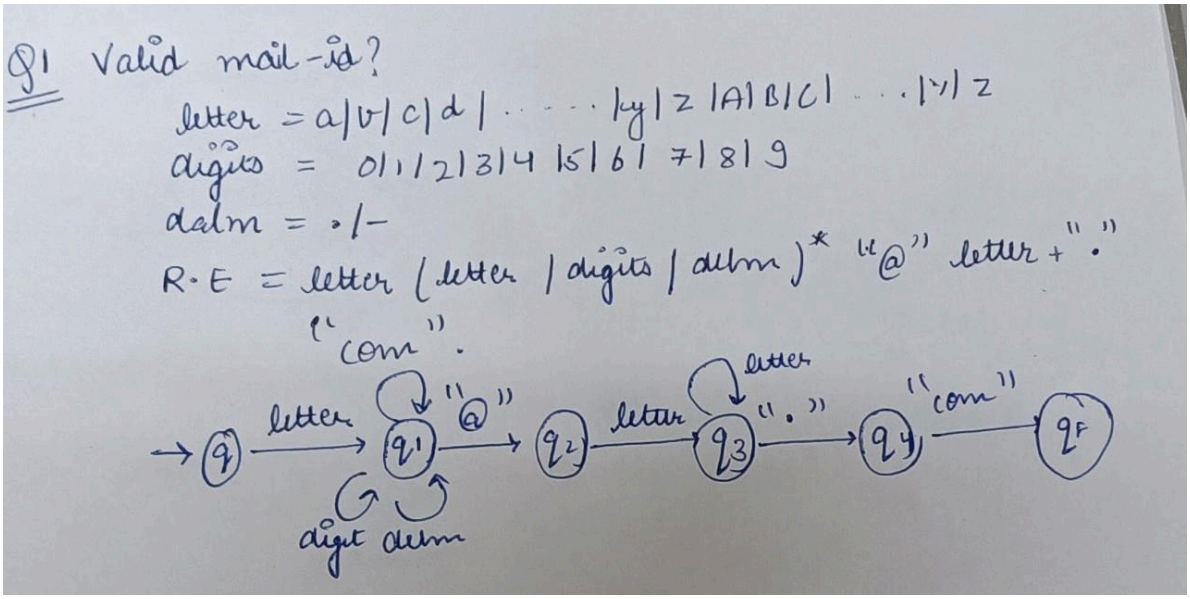
Name- Sparsh garg
Roll No.- Btech/25061/19
SEM- Cse 6th sem

1. Program to check valid mail ID.

```
#include<stdio.h>
#include<string.h>
#include<ctype.h>  int
main(){
char str[1000];
printf("Enter email id: ");
scanf("%s", str);
int n = strlen(str);

int f0=0, f1=0, ans = 1, ct=0, p=0, cp=0;
char c;
for(int i=0; i<n; i++){
if(str[i]==' '){
break;

}else{
```

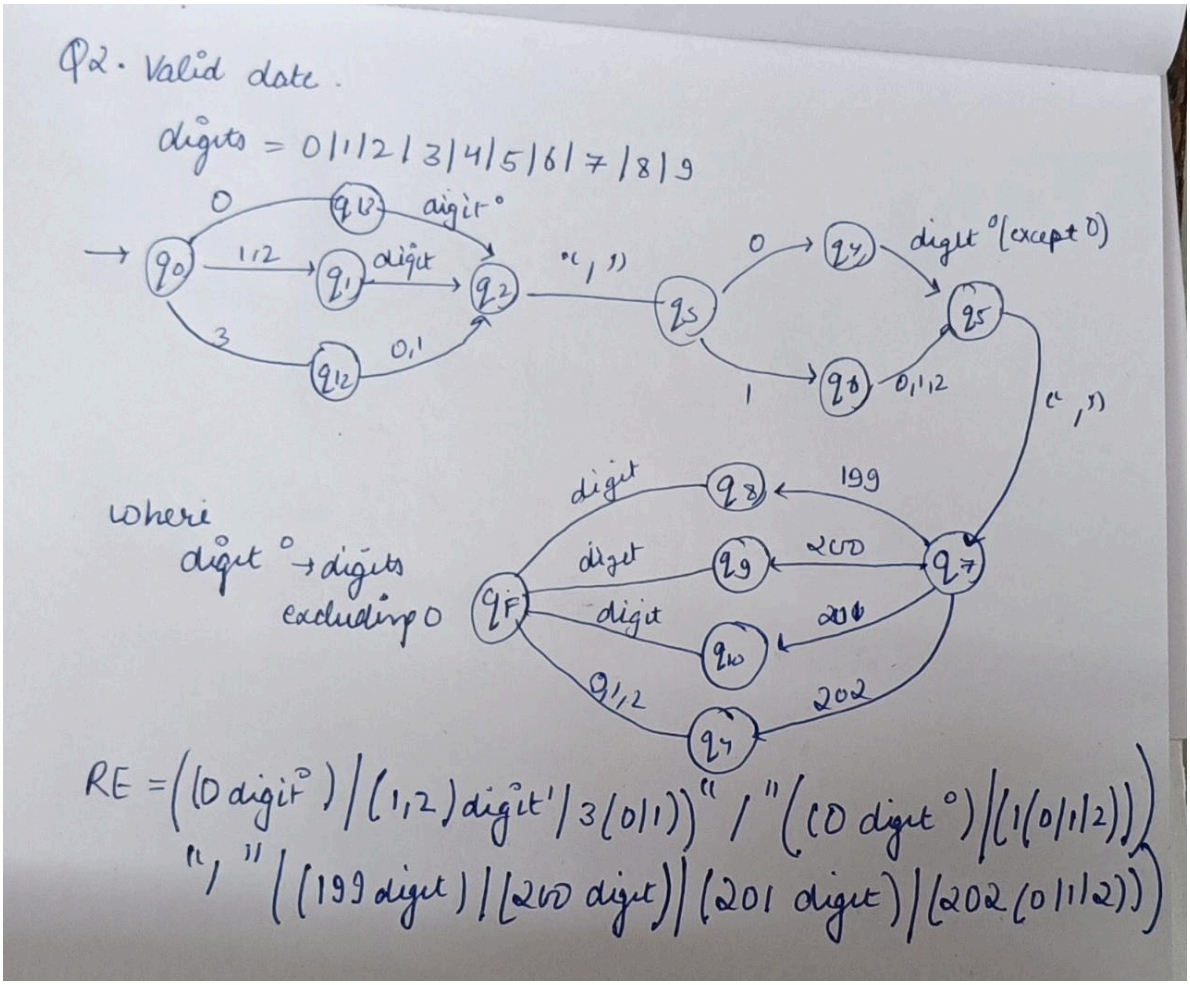
```
c = str[i];
if(isalpha(c)){
continue;
}else if(c == '@'){ p
= i;
f0 = 1;

ct++;

}else if(c == '.'){
if(str[i-1]=='.' || str[i+1]=='.' || str[i-1]=='@' ||
str[i+1]=='@'){ break;
}
```

```
if(i>p){
f1 = 1;
cp++;

}

}else{
continue;
}
}
}
}

if(ct==1 && f0==1 && f1==1 &&
cp==1){ printf("Valid email-id");
}else {
printf("Invalid email-id");

}

}
```



2. Program to check valid date.

```
#include<stdio.h>
int main(void)
{

int d,m,y;
int flag=1,isleap=0;

printf("Enter date(dd/mm/yyyy): ");
scanf("%d/%d/%d",&d,&m,&y);
if(y%100!=0 && y%4==0 ||
y%400==0) isleap=1;

if(y<1850 || y>2050 || m<1 || m>12 || d<1 || d>31)
flag=0;
else if(m==2)        {

if(d==30 || d==31 || (d==29 && !isleap)
) flag=0;
```




```
}

 else if(m==4 || m==6 || m==9 || m==11){
 if(d==31)
 flag=0;

 }


 if(flag==0)
 printf("Not a valid
 date\n"); else
 printf("Valid Date
```

```
\n"); return 0;
}
```

3. Program to check C token.

```
#include <stdbool.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
bool isDelimiter(char ch) {
if (ch == ' ' || ch == '+' || ch == '-' || ch ==
'*' || ch == '/' || ch == ',' || ch == ';' || ch
== '>' ||
ch == '<' || ch == '=' || ch == '(' || ch ==
')' || ch == '[' || ch == ']' || ch == '{' ||
ch == '}') return (true);
return (false);

}
bool isOperator(char ch) {

if (ch == '+' || ch == '-' || ch ==
'*' || ch == '/' || ch == '>' || ch
== '<' || ch == '=')
return (true);
return
(false);
```

```
bool validIdentifier(char* str) {

 if (str[0] == '0' || str[0] == '1' || str[0] == '2' ||

 str[0] == '3' || str[0] == '4' || str[0] == '5' ||
 str[0] == '6' || str[0] == '7' || str[0] == '8' || str[0]
 == '9' || isDelimiter(str[0]) == true)
 return
 (false);
 return (true);
 }
 bool isKeyword(char* str) {

 if (!strcmp(str, "if") || !strcmp(str, "else") ||

 !strcmp(str, "while") || !strcmp(str, "do") ||
 !strcmp(str, "break") ||
 !strcmp(str, "continue") || !strcmp(str, "int")

 || !strcmp(str, "double") || !strcmp(str, "float")

 || !strcmp(str, "return") || !strcmp(str, "char")
 || !strcmp(str, "case") || !strcmp(str, "char")
 || !strcmp(str, "sizeof") || !strcmp(str, "long")

 || !strcmp(str, "short") || !strcmp(str, "typedef")

 || !strcmp(str, "switch") || !strcmp(str, "unsigned")
 || !strcmp(str, "void") || !strcmp(str, "static")
 || !strcmp(str, "struct") || !strcmp(str,
 "goto")) return (true);
 return (false);
```

```
}
bool isInteger(char*
str) { int i, len =
strlen(str);
if (len == 0)
return
(false);
for (i = 0; i < len; i++) {
if (str[i] != '0' && str[i] != '1' && str[i] != '2'

&& str[i] != '3' && str[i] != '4' && str[i] != '5' && str[i]
!= '6' && str[i] != '7' && str[i] != '8'
&& str[i] != '9' || (str[i] == '-' && i
> 0)) return (false);
}
```

```
return (true);

}

bool isRealNumber(char*
str) { int i, len = strlen(str);
bool hasDecimal =
false; if (len == 0)
return (false);
for (i = 0; i < len; i++) {
if (str[i] != '0' && str[i] != '1' && str[i] != '2'

&& str[i] != '3' && str[i] != '4' && str[i] != '5'

&& str[i] != '6' && str[i] != '7' && str[i] != '8'
&& str[i] != '9' && str[i] != '.' ||
(str[i] == '-' && i >
0)) return (false);
if (str[i] == '.')
hasDecimal = true;
}

return (hasDecimal);

}
char* subString(char* str, int left, int
right) { int i;
char* subStr = (char*)malloc( sizeof(char) * (right -
left + 2)); for (i = left; i <= right; i++)
subStr[i - left] = str[i];
subStr[right - left + 1] =
```

```
'\0'; return (subStr);
}

void parse(char* str)
{ int left = 0, right =
0; int len =
strlen(str);
while (right <= len && left <=
right) { if (isDelimiter(str[right])
== false) right++;
if (isDelimiter(str[right]) == true && left ==
right) { if (isOperator(str[right]) == true)
```

```
printf("'%c' IS AN OPERATOR\n",
str[right]); right++;
left = right;

} else if (isDelimiter(str[right]) == true && left != right

|| (right == len && left != right)) {

char* subStr = subString(str, left,
right - 1); if (isKeyword(subStr) ==
true)
printf("'%s' IS A KEYWORD\n", subStr); else if
(isInteger(subStr)
== true)

printf("'%s' IS AN INTEGER\n",
subStr); else if (isRealNumber(subStr)
== true) printf("'%s' IS A REAL
NUMBER\n", subStr); else if
(validIdentifier(subStr) == true
&& isDelimiter(str[right - 1]) == false)
printf("'%s' IS A VALID
IDENTIFIER\n", subStr); else if
(validIdentifier(subStr) == false
&& isDelimiter(str[right - 1]) == false)

printf("'%s' IS NOT A VALID
IDENTIFIER\n", subStr); left = right;
}
}
return;
```

```
}

int main() {
char str[100] = "int p = q *
r;"; parse(str);
return (0);

}
```