

EXP 1:

Code:

```
#include <stdio.h>

#include <string.h>

int main() {    char str[10];

    printf("Enter string: ");

    scanf("%s", str);

    int len = strlen(str);

    if (len >= 2 && str[0] == 'a' && str[len - 1] == 'a')

        printf("Accepted\n");    else

        printf("Rejected\n");
```

EXP 2:

Code:

```
#include <stdio.h>

#include <string.h>

int main() {

    char str[10];

    printf("Enter string: ");

    scanf("%s", str);

    int len = strlen(str);

    if (len >= 2 && str[0] == '0' && str[len - 1] == '1')

        printf("Accepted\n");    else

        printf("Rejected\n");

}
```

EXP 3:

```
#include <stdio.h>

#define MAX 10

int n, eps[MAX][MAX], visited[MAX];

void eClosure(int state) {

    visited[state] = 1;

    for (int i = 0; i < n; i++)

        if (eps[state][i] && !visited[i])

            eClosure(i);

}

int main() {

    printf("Enter number of states: ");

    scanf("%d", &n);

    printf("Enter  $\epsilon$ -move matrix:\n");

    for (int i = 0; i < n; i++)

        for (int j = 0; j < n; j++)

            scanf("%d", &eps[i][j]);

    for (int i = 0; i < n; i++) {
```

```

for (int j = 0; j < n; j++) visited[j] = 0;

printf("ε-closure(q%d) = { q%d", i, i);

eClosure(i);

for (int j = 0; j < n; j++)

    if (visited[j] && j != i) printf("

q%d", j);

printf(" }\n");

} }

```

EXP 4:

Code:

```

#include <string.h> int main() {    char str[10];
printf("Enter string: ");    scanf("%s", str);    int len =
strlen(str), valid = 1;    if (len >= 2 && str[0] == '0' &&
str[len - 1] == '1') {        for (int i = 1; i < len - 1; i++)
if (str[i] != '0' && str[i] != '1') { valid = 0; break; }
        } else valid = 0;    printf(valid ?
"Accepted\n" : "Rejected\n");
}

```

EXP 5:

Code:

```
#include <stdio.h>

#include <string.h> int main() {
    char str[10];
    printf("Enter string: ");
    scanf("%s", str);
    int len = strlen(str), pal = 1;
    for (int i = 0; i < len / 2; i++)
        if (str[i] != str[len - 1 - i]) { pal = 0; break; }    printf(pal
? "Accepted (Palindrome)\n" : "Rejected\n");
}
```

EXP 6:

Code:

```
#include <string.h> #include <stdbool.h>

bool isA(char *str, int *i) {    while (str[*i]
== '0' || str[*i] == '1') (*i)++;    return 1; // ε
possible, so always true
} bool isS(char *str)
{
    int i = 0;    if (!isA(str, &i)) return 0;    if
(str[i++] != '0' || str[i++] != '0') return 0;    if
(!isA(str, &i)) return 0;    return str[i] ==
'\0';
} int main() {    char str[50];
printf("Enter string: ");    scanf("%s",
```

```
str);    printf(isS(str) ? "Valid\n" :  
"Invalid\n");  
}
```

EXP 7:

Code:

```
#include <stdio.h>  
  
#include <stdbool.h> #include <string.h>  
  
bool isA(char *str, int *i) {    while (str[*i]  
== 'a' || str[*i] == 'b') (*i)++;  
    return 1;  
} bool isS(char *str) {  
    int i = 0;  
    if (str[i++] != 'a') return 0;  
    isA(str, &i);  
    if (str[i++] != 'b') return 0;  
    return str[i] == '\0';  
}  
int main() {  
    char str[50];  
    printf("Enter string: ");  
    scanf("%s", str);  
    printf(isS(str) ? "Valid\n" : "Invalid\n");  
}
```

} **EXP**

8:

Code:

```
#include <stdio.h>

#include <string.h>

#include <stdbool.h> bool

checkNFA(char *str) {
    int len = strlen(str);    if (len < 4) return 0;
return (str[0] == '0' && str[1] == '0' &&
str[len-2] == '1' && str[len-1] == '1');
} int main() {    char
str[50];    printf("Enter
string: ");    scanf("%s",
str);
printf(checkNFA(str) ?
"Accepted\n" :
"Rejected\n");

}
```

EXP 9:

Code:

```
#include <stdio.h>

#include <stdbool.h> bool isA(char *str, int
*i) {    while (str[*i] == 'a' || str[*i] == 'b')
(*i)++;    return 1; } bool isS(char *str) {
    int i = 0;    isA(str, &i);
if (str[i++] != 'a') return 0;
isA(str, &i);    if (str[i++]
!= 'a') return 0;
```

```

    isA(str, &i);    return str[i]
== '\0';
} int main() {    char str[50];
printf("Enter string: ");    scanf("%s",
str);    printf(isS(str) ? "Valid\n" :
"Invalid\n");
}

```

EXP 10:

Code:

```

#include <stdio.h>

#include <stdbool.h> #include
<string.h> bool isA(char *str) {
for (int i = 0; str[i]; i++)    if
(str[i] != 'a' && str[i] != 'b')
return 0;    return 1; // ε also valid
} int main() {    char str[50];
printf("Enter string: ");    scanf("%s",
str);    printf(isA(str) ? "Valid\n" :
"Invalid\n");
}

```

EXP 11

code :

```
#include <stdio.h>
#include <string.h>
char str[50];
int i = 0;
int S() {
    if (str[i] == 'a') {
        i++;
        if (!S()) return 0;
        if (str[i] == 'b') { i++; return 1; }
        else return 0;
    }
    return 1;
}
int main() {
    printf("Enter string: ");
    scanf("%s", str);
    if (S() && i == strlen(str))
        printf("String is accepted\n");
    else
        printf("String is rejected\n");
}
```

EXP 12:

Code:

```
#include <stdio.h>
#include <string.h>
int main() {
    char s[5];
    printf("Enter string: ");
    scanf("%s", s);
    int len = strlen(s);
    if (len >= 2 && s[len-2] == 'a' && s[len-1] == 'b')
        printf("Accepted\n");
    else
        printf("Rejected\n");
}
```


EXP 13:**Code:**

```
#include <stdio.h>
#include <string.h>
int main() {
    char s[50];
    printf("Enter string: ");
    scanf("%s", s);
    if (strstr(s, "01"))
        printf("Accepted\n");
    else
        printf("Rejected\n");
}
```

EXP 14:**Code:**

```
#include <stdio.h>
#include <string.h>
int main() {
    char s[5];
    int i=0;
    printf("Enter string: ");
    scanf("%s", s);
    while (s[i]!='a') i++;
    if (i==0) { printf("Rejected\n"); return 0; }
    int bCount=0;
    while (s[i]!='b') { i++; bCount++; }
    if (bCount>0 && s[i]=='\0') printf("Accepted\n");
    else printf("Rejected\n");
}
```

EXP 15:**Code:**

```
#include <stdio.h>
#include <string.h>
int main() {
    char s[5];
    printf("Enter string: ");
    scanf("%s", s);
    int i, n = strlen(s), flag=1;
    for (i=0; i<n/2; i++)
        if (s[i] != s[n-i-1]) { flag=0; break; }
    printf(flag ? "Palindrome\n" : "Not Palindrome\n");
}
```

Exp 16:**Code:**

```
#include <stdio.h>
#include <string.h>
int main() {
    char s[5];
    int state=0;
    printf("Enter binary string: ");
    scanf("%s", s);
    for (int i=0; s[i]; i++)
        if (s[i]!='0') state = !state;
    printf(state==0 ? "Accepted\n" : "Rejected\n");
}
```

Exp 17:**Code:**

```
#include <stdio.h>
int main() {
    char s[5];
    int a=0,b=0;
    printf("Enter string: ");
    scanf("%s", s);
    for (int i=0; s[i]; i++) {
        if (s[i]=='a') a++;
        else if (s[i]=='b') b++;
    }
    printf("a=%d b=%d\n", a, b);
}
```

EXP 18:**Code:**

```
#include <stdio.h>
#include <string.h>

int main() {
    char s[5];
    printf("Enter string: ");
    scanf("%s", s);
    int i=0, flag=1;
    while (s[i]) {
        if (s[i]=='a' && s[i+1]=='b') i+=2;
        else { flag=0; break; }
    }
    printf(flag ? "Accepted\n" : "Rejected\n");
}
```

EXP 19:**Code:**

```
#include <stdio.h>
#include <string.h>

int main() {
    char s[5];
    printf("Enter string: ");
    scanf("%s", s);
    int i=0;
    while (s[i]=='a') i++;
    if (s[i]=='b' && s[i+1]=='\0') printf("Accepted\n");
    else printf("Rejected\n");
}
```

EXP 20:**Code:**

```
#include <stdio.h>
```

```
int main() {  
    char s[50];  
    int state=0; // remainder mod 3  
    printf("Enter binary string: ");  
    scanf("%s", s);  
    for (int i=0; s[i]; i++) {  
        int bit = s[i]-'0';  
        state = (state*2 + bit) % 3;  
    }  
    printf(state==0 ? "Accepted\n" : "Rejected\n");  
}
```