

LAB ASSINGMENT 7

Removing Duplicates: Given a string 'str', remove all adjacent duplicate alphabet symbols present in it by retaining one as the representative.

Ex 1: str = 'aaaabbbbbccccdeeeee';

output : 'abcde'

Ex 2: str = 'xyzz';

output : 'xyz'

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

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

```
int main()
```

```
{
```

```
    char str[30]= "beepboopbeep";
```

```
    int length= strlen(str);
```

```
    for(int i=0; i<length; i++)
```

```
    {
```

```
        char ch = str[i];
```

```
        //Check if current character matches any other character in  
        the subsequence
```

```
        for(int j=i+1; j<length; ){
```

```
            if(str[i] == str[j]){
```

```

        //If yes, then shift the right characters to the left
        for(int k=j; k<length; k++){
            str[k] = str[k+1];
        }
        length--;

    } else {
        //only increment if the duplicate is not found
        //because after the shift, index j can again have duplicate
        j++;
    }
}

printf("%s",str);
return 0;
}

```

| main.c | Run | Output |
|---|-----|--------------------------------------|
| <pre> 1 #include <stdio.h> 2 #include <string.h> 3 4 int main() 5 { 6 char str[30]= "beepboopbeep"; 7 int length= strlen(str); 8 9 for(int i=0; i<length; i++) 10 { 11 char ch = str[i]; 12 13 //Check if current character matches any other character in the subsequence 14 for(int j=i+1; j<length;){ 15 16 if(str[i] == str[j]){ 17 //If yes, then shift the right characters to the left 18 for(int k=j; k<length; k++){ </pre> | | <pre> /tmp/wsHTXmmRAP.o beepo </pre> |