

Code

```
1  #include<iostream>
2  using namespace std;
3
4
5
6  // Transpose of a matrix
7
8
9
10 // approach -1
11 bool getpairsum(int arr[],int n,int target){
12     for(int i=0; i<n-1; i++){
13         for(int j=i+1; j<n; j++){
14             if(arr[i]+arr[j]==target){
15                 return true;
16             }
17         }
18     }
19     return false;
20 }
21
22
23 // approach -2
24 bool getpairsum2(int arr[],int n, int target){
25
26     int low=0;
27     int high=n-1;
28
29     while(low<high){
30         // step-1
31         if(arr[low]+arr[high]==target){
32             return true;
33         }
34         // step-2
35         else if(arr[low]+arr[high]>target){
36             high--;
37         }
38         else{
39             low++;
40         }
41     }
42     return false;
43 }
44
```

```

45
46 int main()
47 /*
48 10 20 30 10 40 70
49 40 50 60 20 50 80
50 70 80 90 30 60 90
51 |
52
53 */
54
55 // transpose of a matrix
56
57 int m[3][3]={10,20,30},{40,50,60},{70,80,90}};
58
59 for(int i=0; i<3; i++){
60     for(int j=1; j<3; j++){
61         // swap
62         int t=m[i][j];
63         m[i][j]=m[j][i];
64         m[j][i]=t;
65     }
66 }
67
68 // print
69 for(int i=0; i<3; i++){
70     for(int j=0; j<3; j++){
71         cout<<m[i][j]<<" ";
72     }
73     cout<<endl;
74 }
75
76 // int main(){
77 //
78 // // int arr[100];
79 // int n;
80 // cin>>n;
81 // int arr[n];
82 // for(int i=0; i<n; i++){
83 //     cin>>arr[i];
84 // }
85 //
86 // for(int i=0; i<n; i++){
87 //     for(int j=i; j<n; j++){
88 //         // print i to j
89 //         for(int k=i; k<=j; k++){
90 //             cout<<arr[k]<<" ";
91 //         }
92 //         cout<<endl;
93 //     }
94 // }
95 // }
96 // }
97
98
99

```

```

void remalloccofchar(char arr[],char c){

    // len arr
    int len=strlen(arr);
    int i; // for traversing the arr
    int j=0; // for mainting the index of final arr

    for(i=0; i<len; i++){
        if(arr[i]!=c){
            arr[j]=arr[i];
            j++;
        }
    }
    // null char at the end
    arr[j]='\0';
}

```

```

90
91 void removedub(char arr[]){
92
93     // find len
94     int len=strlen(arr);
95     int i=1;
96     int j=1;
97
98     char lastchar=arr[0];
99     while(i<len){
100         if(arr[i]!=lastchar){
101             arr[j++]=arr[i];
102             // last char update
103             lastchar=arr[i];
104         }
105         i++;
106     }
107     // place null char
108     arr[j]='\0';
109
110 }

```

```
129
130 // return char jo max
131 char gethighfreq(char arr[]){
132
133     // length
134     int len=strlen(arr);
135
136     char val;
137     // freq arr of size 256
138     int freq[256]={0};
139     int max=INT_MIN;
140
141     for(int i=0; i<len; i++){
142         int input=arr[i];
143         freq[input]++;
144
145     }
146
147
148     // max find
149     for(int i=0; i<len; i++){
150         int input=arr[i];
151         if(freq[input]>max){
152             // max ko update
153             max=freq[input];
154             value=input[i]; // value me max char usko update kar rahe hai
155         }
156     }
157     return value;
158 }
```