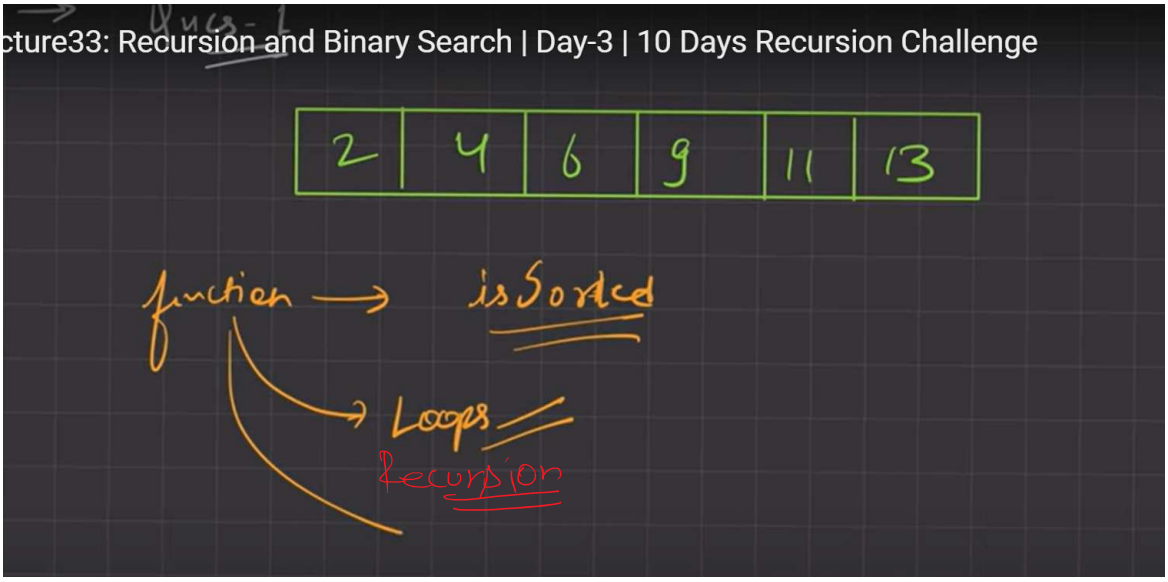
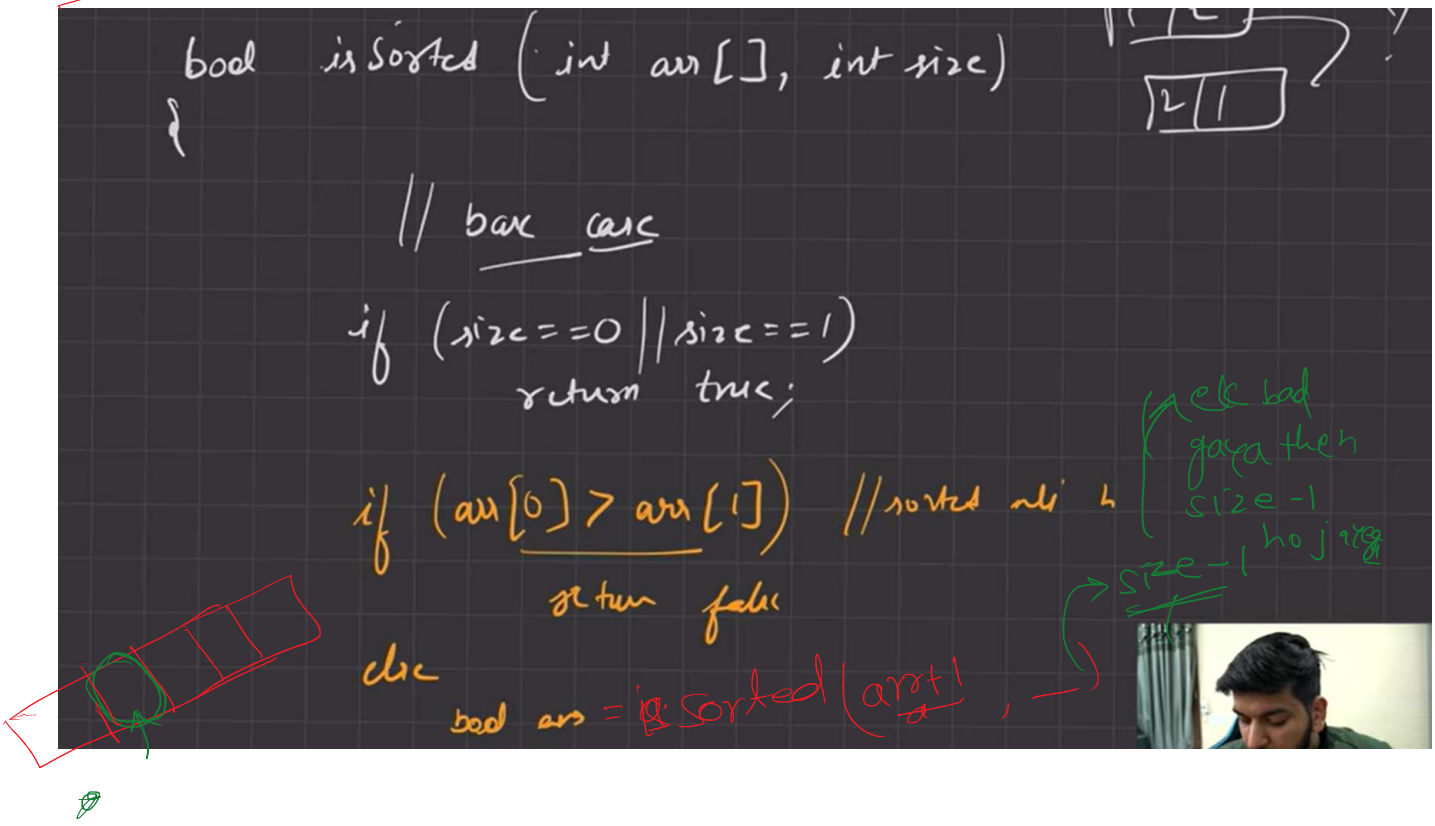
Day3

26 May 2022 17:20

Check is Array is Sorted or not



for



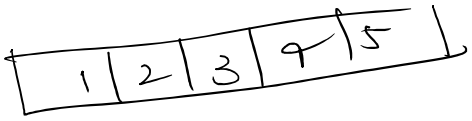
```cpp
#include<iostream>
using namespace std;

bool isSorted(int *arr, int size) {

    //base case
    if(size == 0 || size == 1 ){
        return true;
    }

    if(arr[0] > arr[1])
        return false;
    else {
        bool remainingPart = isSorted(arr + 1, size - 1 );
        return remainingPart;
    }

}

int main() {

    int arr[5] = {2,4,6,8,9};
    int size = 5;

    bool ans = isSorted(arr, size);

    if(ans){
```

arr+1

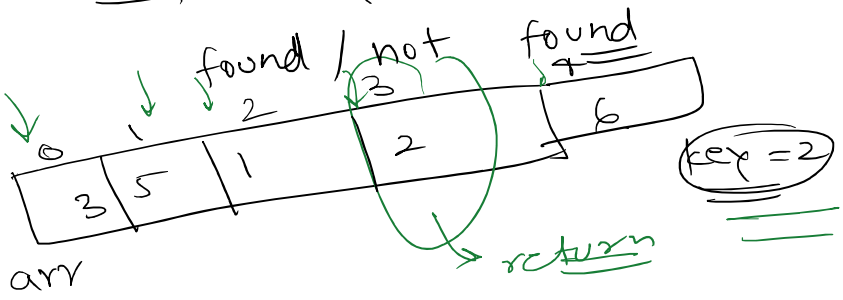
H.W getSum :-


```cpp
int getSum(int *arr, int size) {

    //base case
    if(size == 0) {
        return 0;
    }
    if(size == 1 )
    {
        return arr[0];
    }

    int remainingPart = getSum(arr+1, size-1);
    int sum = arr[0] + remainingPart;

}

int main() {

    int arr[5] = {2,4,9,9,9};
    int size = 5;

    int sum = getSum(arr, size);

    cout << "Sum is " << sum << endl;

    return 0;
```

ques:- linear search :-



array :- key element = 6



int arr[5] = | 3 | 5 | 1 | 2 | 6 |

main int size = 5;
key = 2;
bool ans = linearsearch (arr, size, key)

Base case Agar array khali ho gaya and element nahi mila hai

if (size == 0)
return false;

if (arr[0] == key)
return true;

else
{ bool remainingpart = linearsearch (arr+1, size-1, key)
return remainingpart;
}

| 3 | 5 | 1 | 2 | 6 |

key = 6

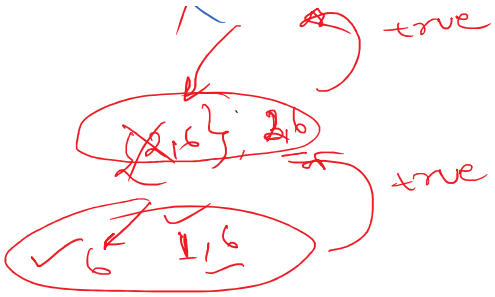
size = 5

{ 3 5 1 2 6 }

{ 5 1 2 6 }, 4, 6 ← size

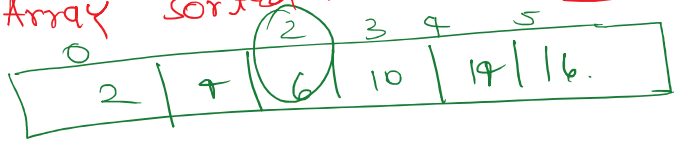
{ 1 2 6 }, 3, 6 ← true

{ 1, 2, 6 }, 2, 6 ← true



if 6 ke jagah eight hota to.

Binary search pe Array sorted hone chahiye



key 14

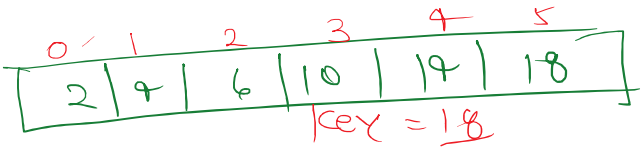
mid = $\frac{0+5}{2}$ = 2 → 6.

14 > mid.

mid $\frac{3+5}{2}$ = 4 → (14)

14 == 14 → return mid

#



$\frac{0+5}{2}$ = 2

mid = 2

arr[2] = 6

if (arr[mid] < key)

Search Right wale half me.

else

search karo left wale half me

search in right half s = mid+1

search in left half e = mid-1

binary search (arr, start, end, key)

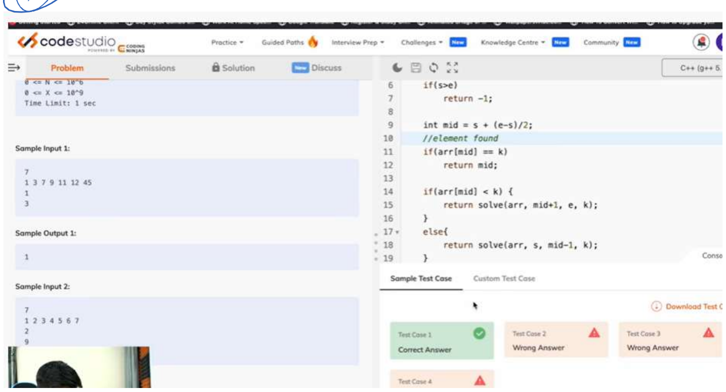
, Right binary (arr, mid+1, end, key)

left binary (arr, start, mid-1, key)

base case ① if (s>e) → return -1 → element not found.

② // element found.
if(arr[mid]==k)
return true;

②



#DSABusted #LoveBabbar

Lecture34: Recursion with Strings | Day-4 | 10 Day Recursion Challenge

Pay4

que1 :- i/p :- "abcde"
o/p :- "edcba"

② i/p :- "babbar"
o/p :- "rabbab"

edcba , n-1

dcdb {

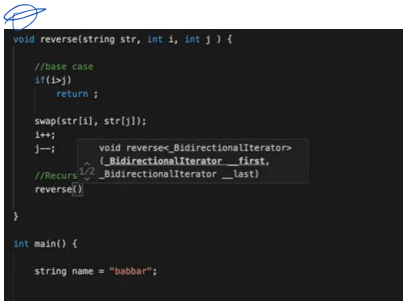
cdb

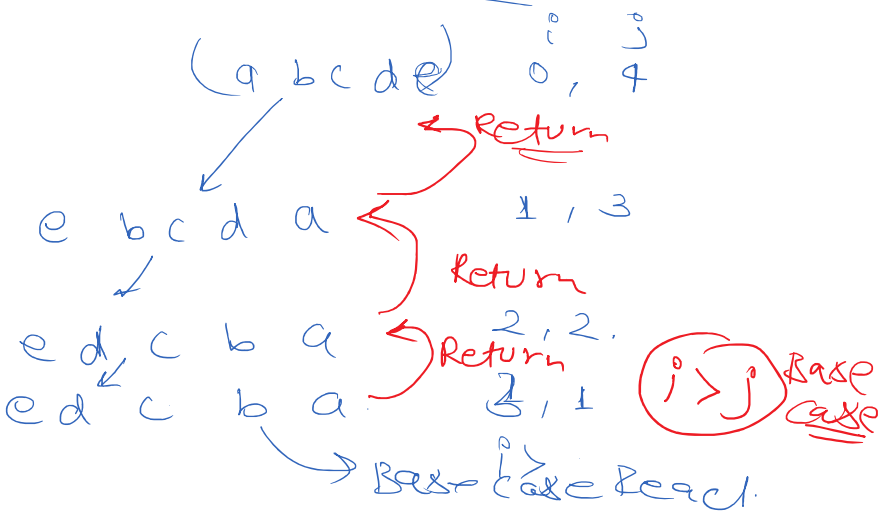
db

if (str.lenth == 0)
return.

reverse(i, j, str).
{
// base case

base case
if (i>j)
return;
swap (s[i], s[j])
i++;
j--
reverse(i, j, str)





que 2 → check palindrome →

string :- abba.

reverse string = abba

string == reverse string

```
bool checkpalindrom ( string str, int i, int j)
{
// Base case.
if(i>j)
    return true;
if (str[i] != str[j])
    return false
// Recursive call-
else return checkPalindrom( str, i+1, j-1);
```

```cpp
#include<iostream>
using namespace std;

bool checkPalindrome(string str, int i, int  j) {

    //base case
    if(i>j)
        return true;

    if(str[i] != str[j])
        return false;
    else{
        //Recursive call
        return checkPalindrome(str, i+1,j-1);
    }
}

int main() {

    string name = "BookkooB";
    cout << endl;

    bool isPalindrome = checkPalindrome(name, 0, name.length()-1 );
```

```
lovebabbar@192 Lecture34:
bar/Documents/CodeHelp-DSA
 Day4/" && g++ palindromeC
ers/lovebabbar/Documents/C
: Recursion Day4/"palindro
palindromeCheck.cpp:8:9:
drome' should return a val
        return ;
1 error generated.
lovebabbar@192 Lecture34:
bar/Documents/CodeHelp-DSA
 Day4/" && g++ palindromeC
ers/lovebabbar/Documents/C
: Recursion Day4/"palindro
Its not a Palindrome
lovebabbar@192 Lecture34:
bar/Documents/CodeHelp-DSA
 Day4/" && g++ palindromeC
ers/lovebabbar/Documents/C
: Recursion Day4/"palindro
Its not a Palindrome
lovebabbar@192 Lecture34:
bar/Documents/CodeHelp-DSA
 Day4/" && g++ palindromeC
ers/lovebabbar/Documents/C
: Recursion Day4/"palindro
Its a Palindrome
lovebabbar@192 Lecture34:
bar/Documents/CodeHelp-DSA
 Day4/" && g++ palindromeC
ers/lovebabbar/Documents/C
: Recursion Day4/"palindro
Its not a Palindrome
```

Que ③ i/p a=3

a^b → b is even → $a(a^{\frac{b}{2}} * a^{\frac{b}{2}})$

b is odd → $2^9 = 2 * 2^4 * 2^4$

$2^9 \rightarrow 2 \times 2^4 \times 2^4 \Rightarrow 2 \times (2^4)^2$

$2^4 \rightarrow 2^2 \times 2^2 \rightarrow (2^2)^2$

$2^2 = 2 \times 2 \Rightarrow (2)^2$

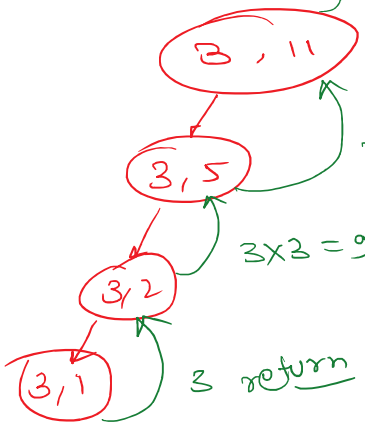
$2^1 = 2 \times 2^0 \times 2^0$

$2^0 = 1$

```cpp
int power(int a, int b) {
    //base case
    if( b == 0 )
        return 1;

    if(b == 1)
        return a;

    //RECURSIVE CALL
    int ans = power(a, b/2);

    //if b is even
    if(b%2 == 0) {
        return ans * ans;
    }
    else {
        //if b is odd
        return a * ans * ans;
    }
}

int main() {

    int a,b;
    cin >> a >> b;
```

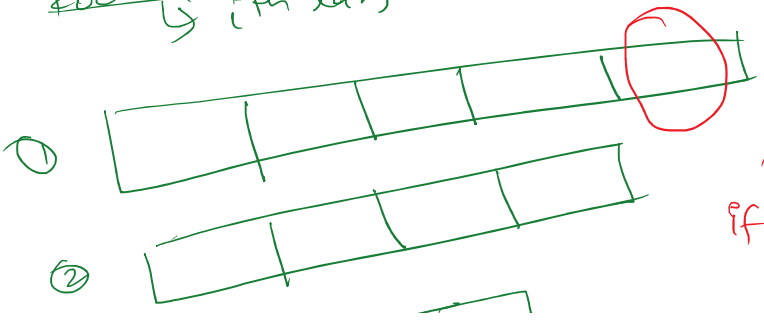
```
/Document
y4/"expon
2 10

Answer is
lovebabba
bar/Docum
 Day4/" &
/Document
y4/"expon
3 10

Answer is
lovebabba
bar/Docum
 Day4/" &
/Document
y4/"expon
3 11
```

$3 \times 243 \times 243 = 177147$



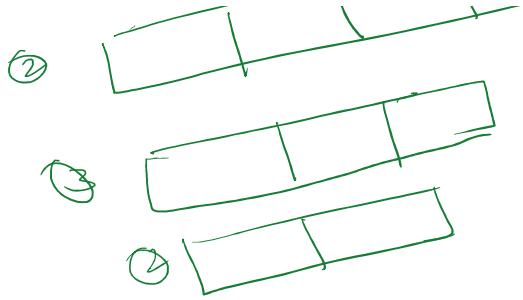
Bubble sort

Round ith : → ith largest (Right place)



Base case

if size == 0 || size == 1
sorted
return;



if size = 0 "sorted"
return;
for(int i=0; i<n-1; i++)
if (arr[i] > arr[i+1])
swap(arr[i], arr[i+1])
}
sortarray (arr, n-1)

```cpp
#include<iostream>
using namespace std;

void sortArray(int *arr, int n) {

    //base case - already sorted
    if(n == 0 || n == 1) {
        return ;
    }

    //1 case sovle karlia - largest element ko end me rakh dega
    for(int i=0; i<n-1; i++) {
        if(arr[i] > arr[i+1]){
            swap(arr[i], arr[i+1]);
        }
    }

    sortArray(arr, n-1);

}

int main() {

    int arr[5] = {2,5,1,6,9};

    sortArray(arr,5);

    for(int i=0; i<5; i++) {
```

HW :- ~~Bubble sort~~
Insertion & selection sort using Recursion