Note Title

COL 100
Sep 13, 2018

13-09-2018

Last class:-

We started:- Vectors / Collections → Data structures

↳ Abstract Data types

```
#include "vector.h"
vector<int> nums;
vector<string> names;
vector<char>
vector<double>
vector<vector>
```

→ C++ STL:- ↳ standard template Library

→ standford Library ↳ "vector.h"

SPL

Arrays (C++ / C)

```
vector<int> nums; → //{ }
vector<int> nums = {1, 2, -3, -2};
                    ↳0 → 1 → 2 → 3
```

→ Capital

Vector<int> nums = {42, 7, -6, 0, 28};

Vector<String> names; // { }

names.add("pritam"); // { "pritam" }

names.add("siya"); // { "pritam", "siya" }

names.insert(0, "Lokesh"); // { "Lokesh", "pritam", "siya" }

Vector

Vector<int> names(5);

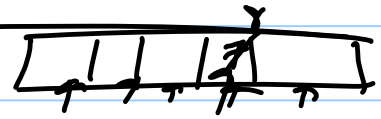
Arrays (vs Vectors) → "fixed size"

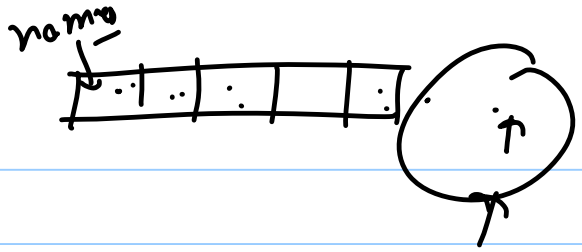
→ int names[5]; → // declaring an array

Array

names.size()

↳ very difficult to add/delete elements / resize





Vector

Array

```
int names[5];
names[7] = 35;  X → Execute
```

↳ Give unexpected behaviour

```
Vector<int> names(5);
names[7] = 30;
```

↳ Clear message (Clear 99)

Vector- member functions

name	description
~~Vector<type> v;~~	
`Vector<type> v;`	→ // initialize a vector of // type data type v // of size 0
`v.add(value);`	→ add value to v (at the end of v).
`v += value;`	
`v += v(val1, val2, ..., valn);`	// add val1, val2, ... valn to v (in order)

Methods

v.clear()

Description

remove all the elements in v. Becomes an empty vector

v[i] or v.get(i)

get the value at index i

↓ int a = 5;

~~int~~ a = v[i]

or v[i] = a;

---

names[5] = "Gautam";

//Error ;

vector<string> names = {"Rakesh", "Sija"};

string str = names[1];

cout << str; // Sija

names[1] = "pritam";

cout << str << names[1];

Sija pritam

"Rakesh"	"pritam
0	1

size 5

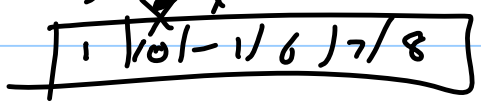
Method →	Description
v.insert(i, value)	→ inserts value at index i.
v.isEmpty()	→ Boolean value → true if vector has no elements false otherwise
v.size()	returns # of elements in the vector

(v.size() == 0)

v.size() // 6

~~v~~er v.remove(i) →

v.size() // 5



v.remove(2);

Member	Description
`v[i] = value;` ⇒ `v.set(i, value);`	set element at position i to value
`v.sublist(start, length)`	→ make a new sub vector starting at position 'start' & of length 'length'
`v.size()`	→ returns size of v
~~list~~ `v.remove(start, length)`	→ write a function ~~for(int i = start; i < start + length; i++)~~ `v.remove(i);`

string str = v.toString()

returns a string representation of v

"[9, 6, -7, 8, 10, 11, 3]"

cout << v;

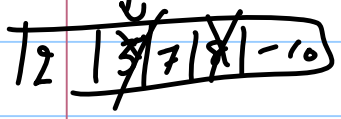
ostr → output stream of a file

print the 'string' representation of vector v

start = 1
length = 3

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

v.remove(1); v.remove(start);

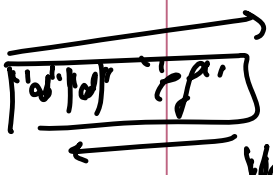


3

X

Homework

Write it using while loop



looping / iterating over all the values of vector

```
vector names{ "ab", "cd", "ef" };
for(int i=0; i<names.size(); i++) {
    cout << names[i] << endl;
}
```

//ab
//cd
//ef

```
for(int i = names.size()-1; i>=0; i--) {
    cout << names[i] << endl;
}
```

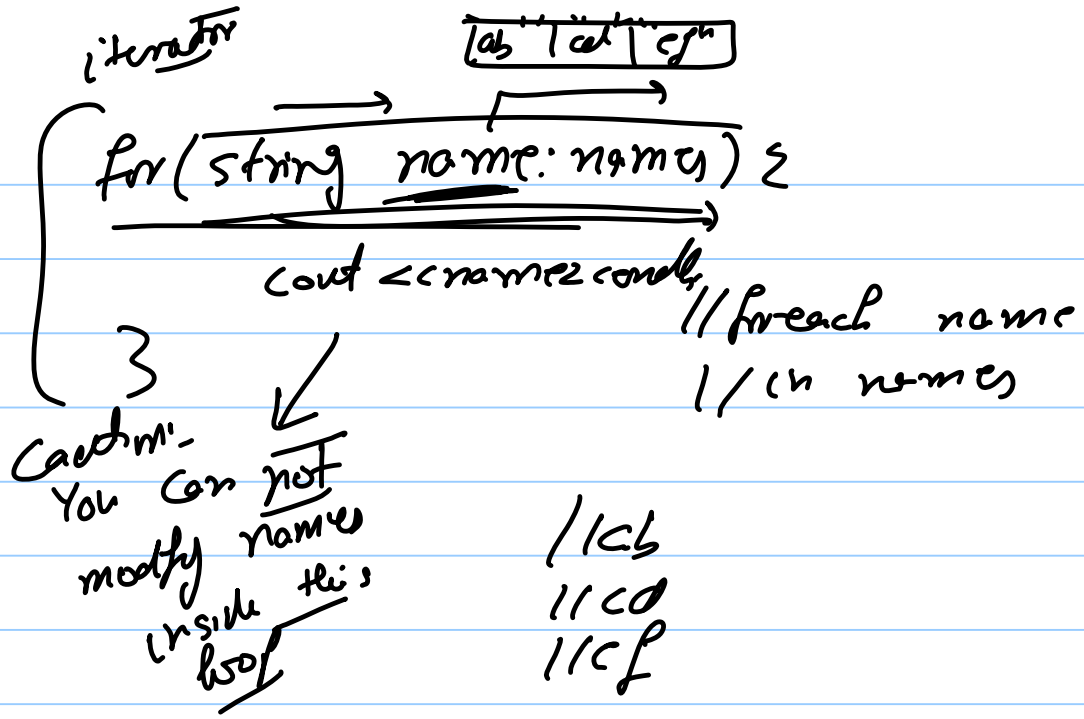
fe | dc | ba

//ef
//cd
//ab

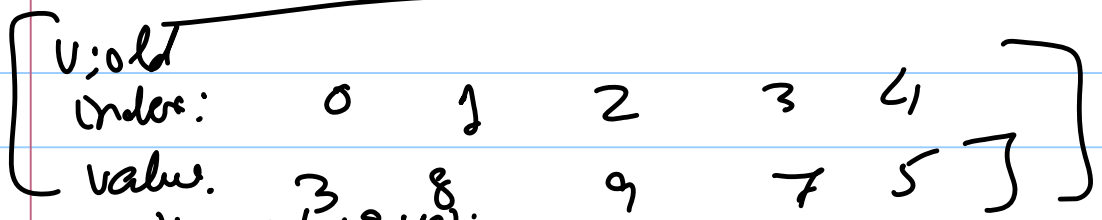
ef cd ab
fe dc ba

```
string str = names.
cout << str
```

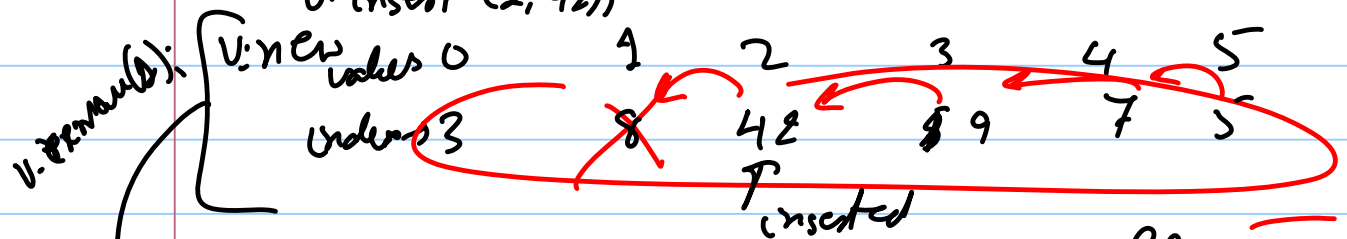
{ "ab", "cd", "ef" }


iterator
"ab" "cd" "ef"
for(string name: names) {
cout << name2 cond;
}
//foreach name
//in names
Caution:
You can not
modify names
inside this
loop
//ab
//cd
//ef

~~v.insert(2, 42)~~

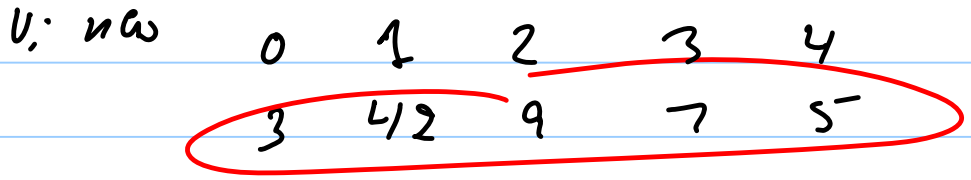


v.insert(2, 42);



efficiency?

v.remove(1);



① Given a vector of strings -

{"ab", "cd", "ef"}

→ write a function reversePrint(Vector<string> vec)

which prints - returns a string

efcdab

② Given a vector of strings

{"ab", "cd", "ef"}

write a function reversePrint(vector<string> v1)

→ Return a string.. fedcba

Write a function ↓ {5, 6, 7
35, -56, 100}

int getMax (Vector<int> vec) {
// returns the max value in vect

Do it
Not Class

3