Introduction to Vectors and Functions

CSE 232 - Dr. Josh Nahum

Table of contents

00

Vectors

Merits of Vectors versus Arrays

Mutable Size

Vectors can change in size and report their size

Run Time Size

The length of the vector can be determined at runtime (instead of compile time).

Member Functions

Vector has useful methods, arrays have none.

Using Vectors

Necessary Library

```
#include <vector>
```

Parameterized Type

```
std::vector<int> nums;
std::vector<double> temperatures;
vector<vector<char>> character_freq;
```

Namespace

```
using std::vector;
// or use std:: prefix
```

push_back

(Instead of Python's append) this method adds an element to the end of the vector.

Indexing

`.at()`

Example: `x.at(2);`
The at member function raises an error if the index is out of bounds.
Good!

`[]`

Example: `x[2];`
The operator[] performs undefined behavior if index is out of bounds.
Bad!

Recommendation:
Always use the at member function instead of operator[].

01 Function Definitions

Definition (and Declaration)

A function must be declared before it can be called.

Example Function Declaration:

```
int add(int, int);
```

Example Function Call:

```
cout << add(4, x);
```

Somewhere in the program the function must be defined exactly once.

Example Function Definition:

```
int add(int a, int b) {
  return a + b;
}
```

Note: A function definition also acts as a declaration.

Functions and type modifiers

For now, please don't declare functions with parameters (or return types) that are pointers, arrays, or references. We will talk about how to use such type modifiers with functions in a few weeks.

02

Live Coding

Vectors with initialization, loops, and multiple functions

Attribution

Please ask questions via Piazza

Dr. Joshua Nahum
www.nahum.us
EB 3504

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**