

CS 32 Bootcamp

02—CS 31 Review

Agenda

- Diagnostic mention: undefined behavior
- Control flow
- Functions
- Vectors
- Strings
- Pointers
- Homework 1

} CS 31

Undefined behavior

- What does undefined behavior mean?
- Undefined behavior
 - `std::cout << arr[-1];` }
- Implementation-defined behavior
 - Size of int is 32 bits
- Throwing an exception
 - File not found

throw Exception()

max = 4

Control flow

max — min

```
for (int i = min; i < max; i++) {  
    foo();  
}
```

3

5

```
while (some_boolean_condition) {  
    foo();  
}
```

```
if (condition_1) {  
    foo();  
} else if (condition_2) {  
    bar();  
} else {  
    baz();  
}
```

```
switch (value) {  
    case 1: —  
    case 2: —  
}
```

Functions

- Function signature

- `int my_function(std::string s, int n);`

- Declaration vs. definition

- Declaration in header (.h) files
 - Definition in source (.cpp) files

↳ why: separate my implementations from the people who are using my functions

• Call/invoke functions in our code

Vectors

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

- Contiguous elements: fast lookup, fast append, slow insertion

$[5, 6, 10, 7, 8, 9]$
A

```
std::vector<int> my_int_vec;  
my_int_vec.push_back(5);  
my_int_vec.push_back(6);  
std::cout << my_int_vec[1];
```

Strings

- Basically a vector of characters
- You can compare strings using `==`
- You can concatenate strings using `+`

"hello"

['h', 'e', 'l', 'l', 'o']

→ operator overloading

string a = "hi"

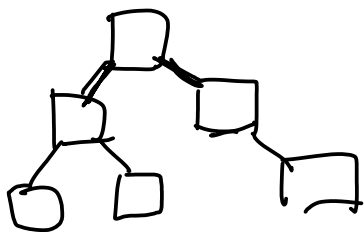
string b = "Hi"

if (a == b) {

c = a + b

c[1] == 'i'

Pointers



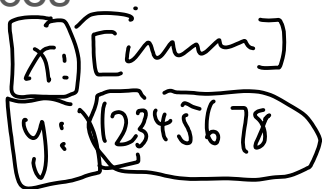
- Address of a variable
- Why do we want pointers?

- We want a function to operate on an existing object
- We want to avoid copying large amounts of data
- We want to connect objects together (e.g. linked list)

- Pointers vs. references

Pointer

```
std::vector<int> x;  
std::vector<int> *y = &x;  
y->push_back(5);  
std::cout << x[0];
```



$\&x \neq \&y$

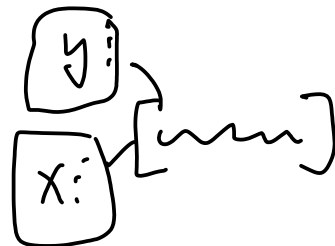
$y == \&x$

```
void capitalize_first(std::string *s_ptr) {  
    // assume the function is implemented  
    s_ptr->[0] = 'H';  
}
```

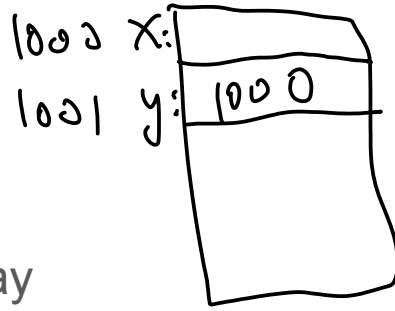
```
...  
std::string s = "hi";  
capitalize_first(&s);  
// &s would be an address 0x01234dead  
s == "Hi" // True
```

Reference

```
std::vector<int> x;  
< std::vector<int> &y = x;  
y.push_back(5);  
std::cout << x[0];
```



Homework 1



- Due this Saturday
- Download skeleton from course website
- Submit on Gradescope
- We will be adding test cases throughout the week, so don't be surprised if the Gradescope changes
- Feel free to ask questions on Discord or during office hours!