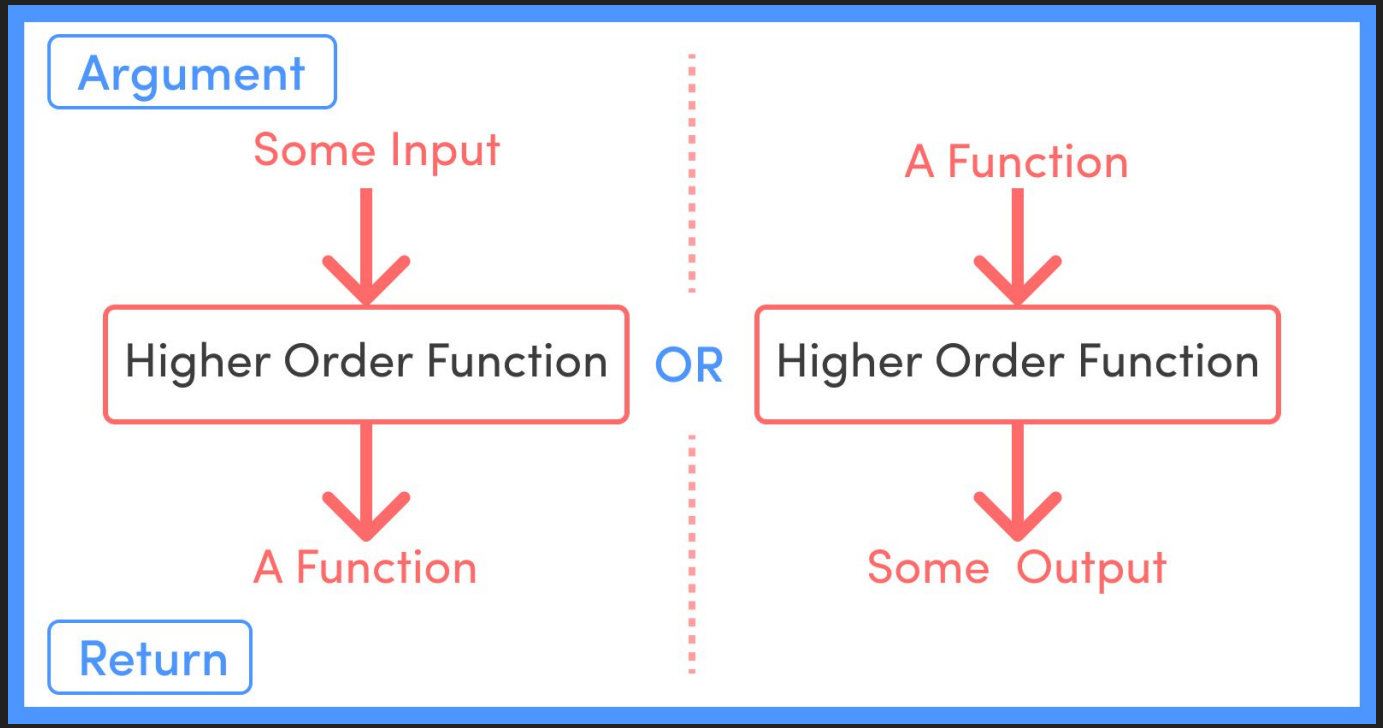
Higher-Order Functions

C++ Edition

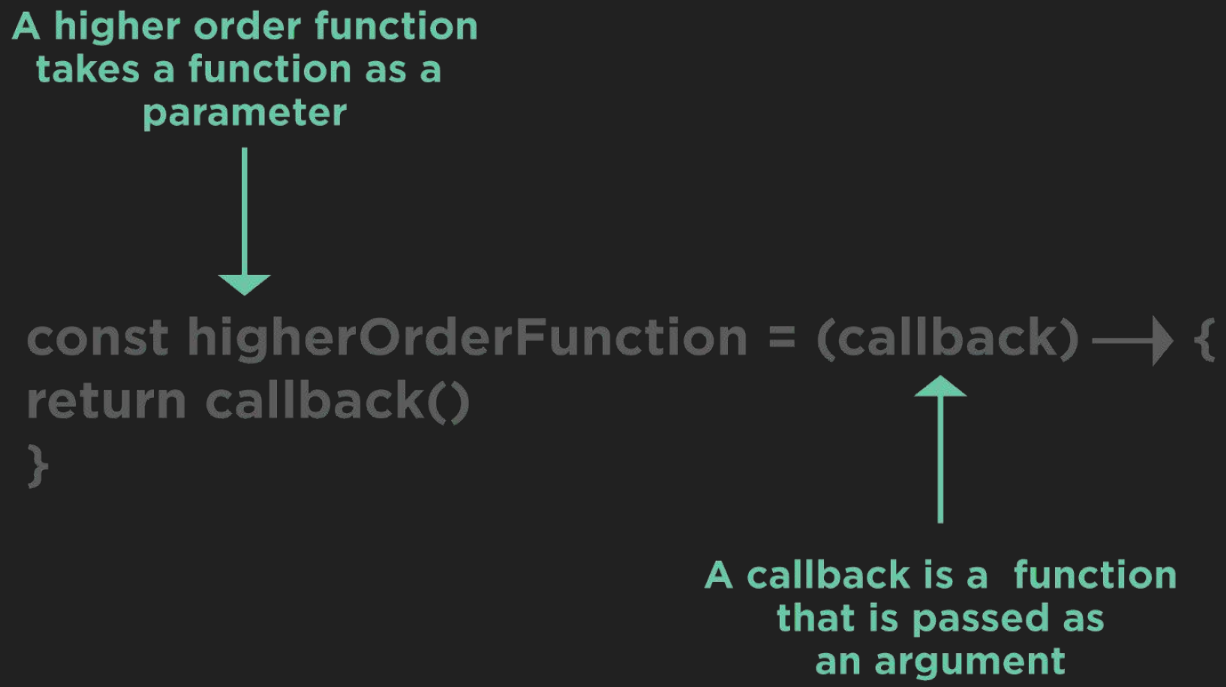
by Tommy Hoang and Jake Davis

What are Higher-order Functions?



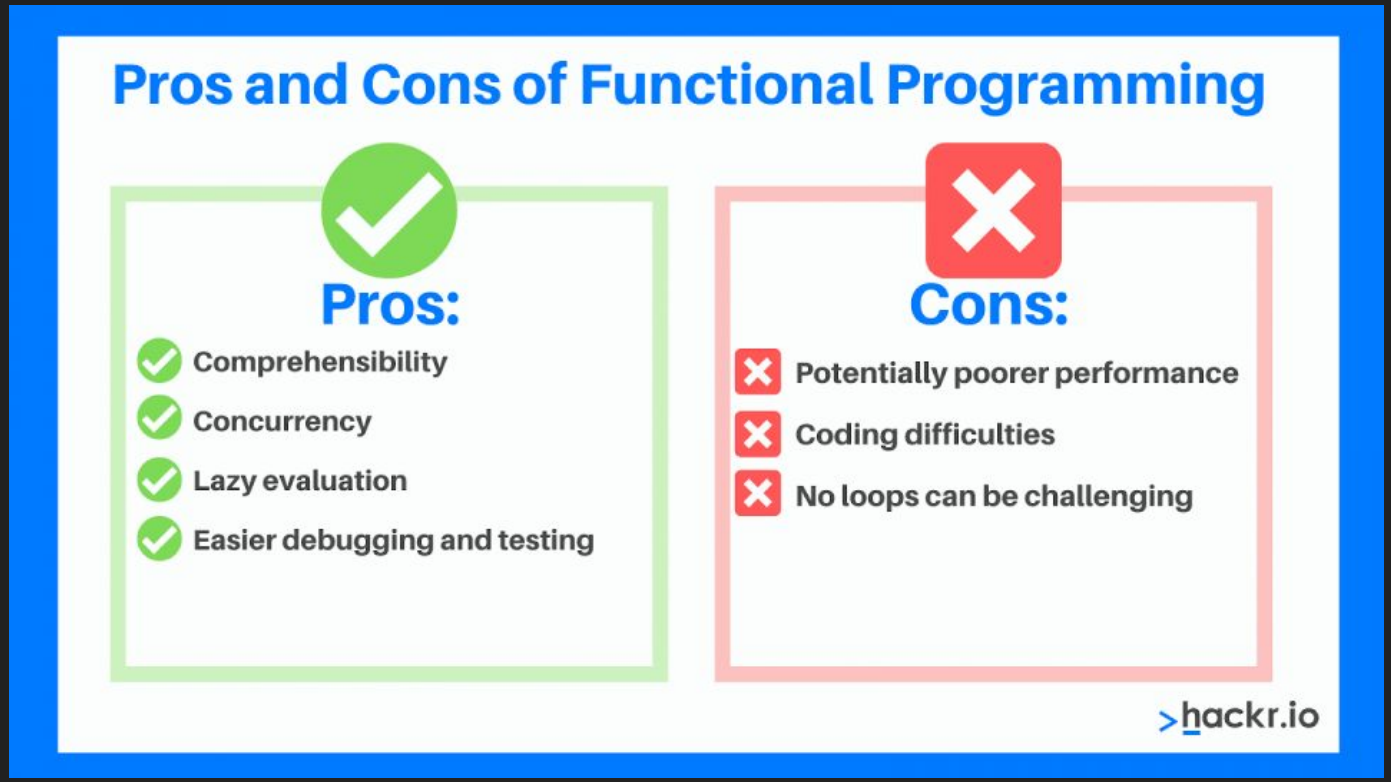
https://www.freecodecamp.org/news/higher-order-functions-in-javascript/

Callback Functions



https://medium.com/@gaathapaul/what-is-a-callback-function-bd6d893f4c59

Functional Programming



https://hackr.io/blog/functional-programming

FUNCTIONS ARE VALUES!!!

Examples From CSCI 3155

```
def foldLeftAndThen[A,B](t: Tree)(z: A)(f: (A,Int) => A)(sc: A => B): B = {
  def loop(acc: A, t: Tree)(sc: A => B): B = t match {
      case Node(l, d, r) => loop(acc, l)((acc) => loop(f(acc, d), r)(sc))
      case Empty => sc(acc)
  }
  loop(z, t)(sc)
}
```

Higher-order Functions in C++?

Workaround

Higher-order Function ->

Lambda ->

```
void Foo()
{
  vector<int> v;
  v.push_back(1);
  v.push_back(2);
  v.push_back(3);
  for_each(begin(v), end(v), [](int i) {
    cout << i << " ";
  });
}
// Outputs:
// 1 2 3
```

https://learn.microsoft.com/en-us/archive/msdn-magazine/2012/august/c-functional-style-programming-in-c

Our Example

```cpp
#include <string>
#include <iostream>
#include <vector>
#include <algorithm>

using namespace std;

int square_plus_one(int n){
    return n*n+1;
}

int main(){
    vector<int> nums{1,2,3,4,5,6,7,8,9};

    //applies square_plus_one to nums and sets it to transform
    transform(nums.begin(),nums.end(),nums.begin(),square_plus_one);

    //prints [2,5,10,17,26,37,50,65,82]
    for(vector<int>::iterator i=nums.begin();i!=nums.end();i++){
        cout<<" "<<*i<<"\n";
    }
}
```

Benefits of Using Higher-order Functions

Thank you for watching!