

Function Call Operator

Exercises

- What is the prototype of the function call operator?
- How is it invoked?
- How is it called?

- What restrictions apply to a programmer writing an overloaded function call operator?

- Explain what a functor is
- What is a functor used for?

- Explain what the class below does
- Implement this class and write a program to exercise it

```
class evenp {  
    public:  
        bool operator() (int n) {  
            return (n % 2 == 0);  
        }  
};
```

- Describe what the code below does
- Why is the second argument passed by value?
- Write a full working program which exercises this code

```
void do_it(const vector<int>& vec, evenp is_even) {  
    for (auto v: vec)  
        if (is_even(v))  
            cout << v << " is even\n";  
}
```

- Explain what it means when a functor has state

- Explain what the class below does
- Implement this class and write a program to exercise it

```
class divisible {  
    private:  
        int divisor {1};  
    public:  
        divisible(int d) : divisor(d) {}  
        bool operator() (int n) {  
            return (n % divisor == 0);  
        }  
};
```