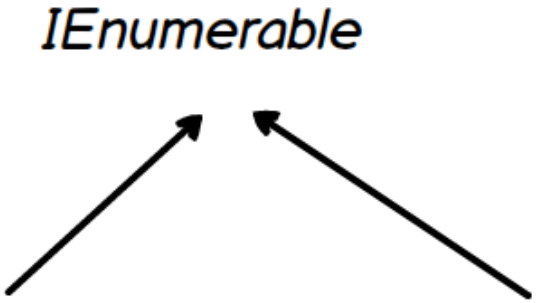
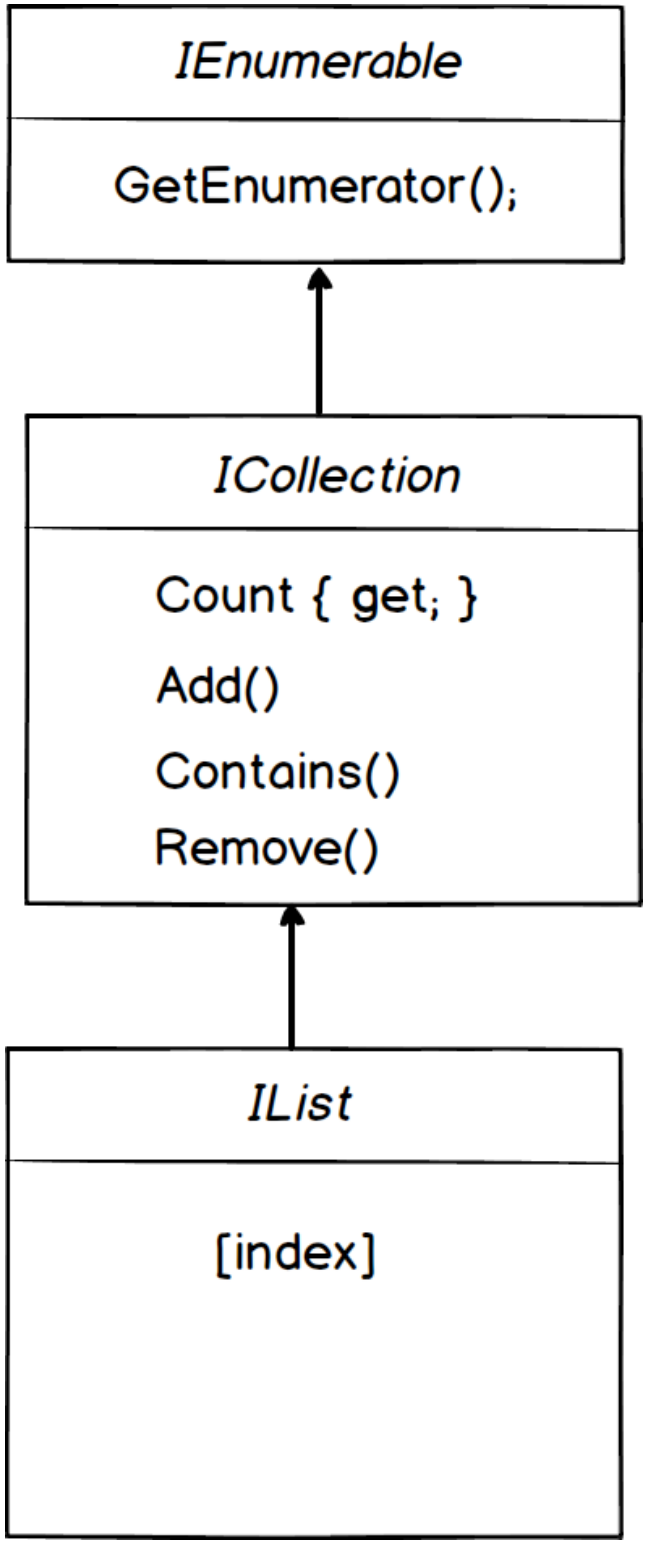
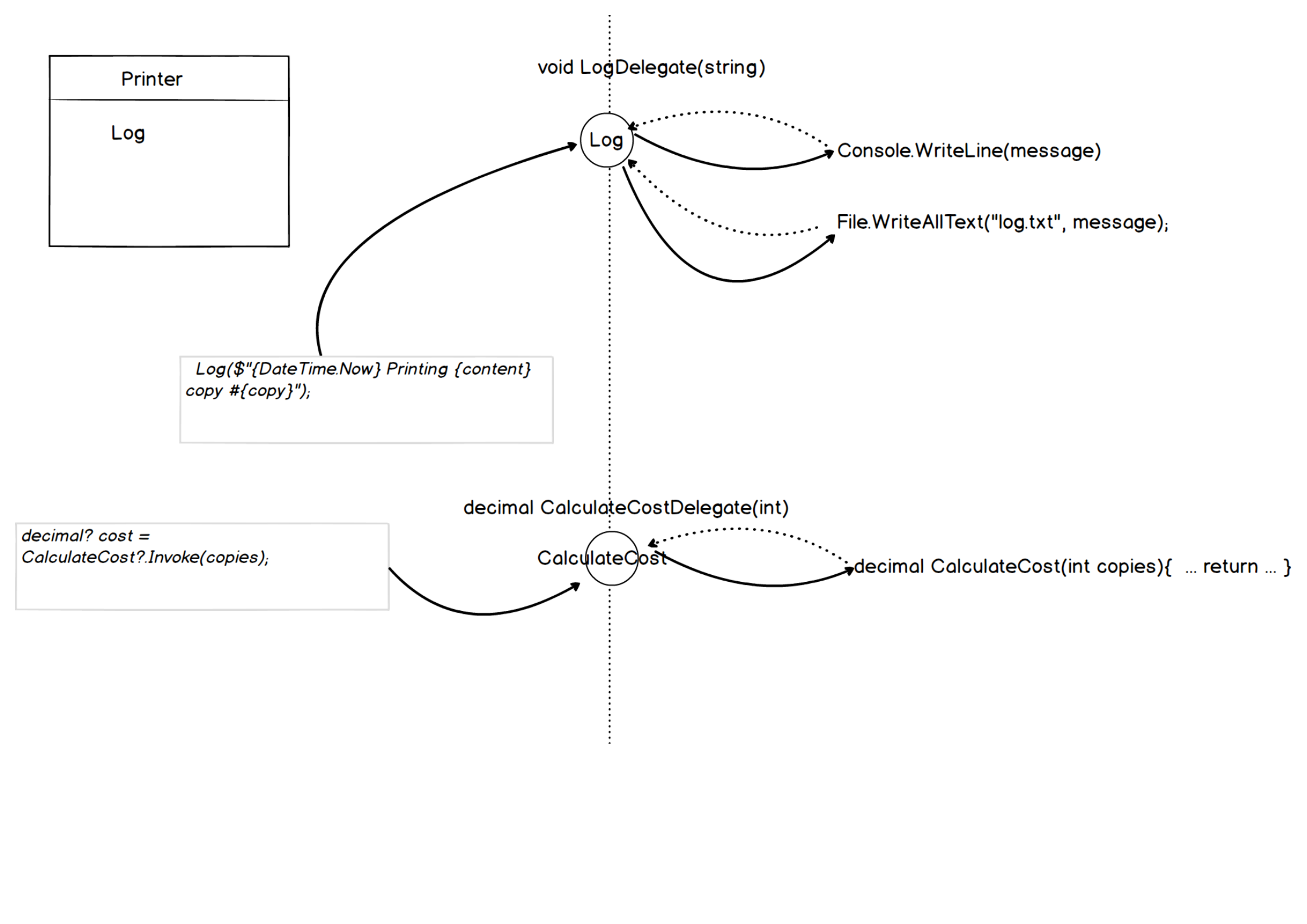



```
int[] numbers = new int[] { 1, 2, 3 }

numbers[1] = 9;
```

```
List<int> numbers = new List<int>();

numbers[1] = 9;

numbers.Add(7);

numbers.Remove(7);
```


IEnumerable
GetEnumerator();
ICollection
Count { get; }
Add()
Contains()
Remove()
IList
[index]


Printer
Log
void LogDelegate(string)
Log
Console.WriteLine(message)
File.WriteAllText("log.txt", message);
Log($"{DateTime.Now} Printing {content}
copy #{copy}");
decimal CalculateCostDelegate(int)
decimal? cost =
CalculateCost?.Invoke(copies);
CalculateCost
decimal CalculateCost(int copies){ ... return ... }

Zapis funkcyjny

f(x) = x + 2

g(x) = x^2 x należy do C

f(x, y) = x + y x i y należy do C

```
int F(int x)
{
  return x + 2;
}
```

```
int F(int x, int y)
{
  return x + y;
}
```

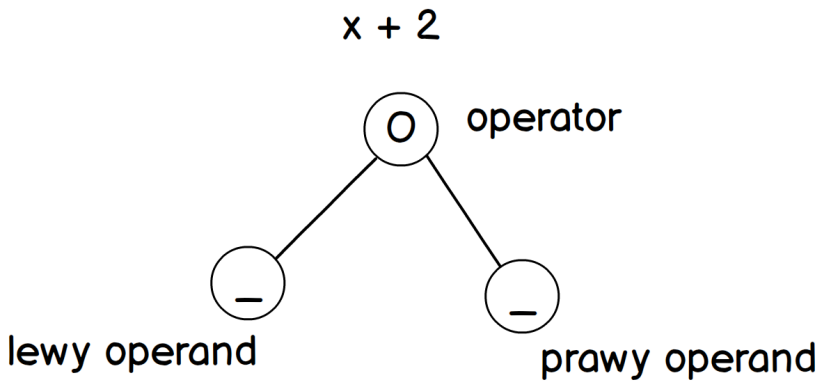
Wyrażenia lambda

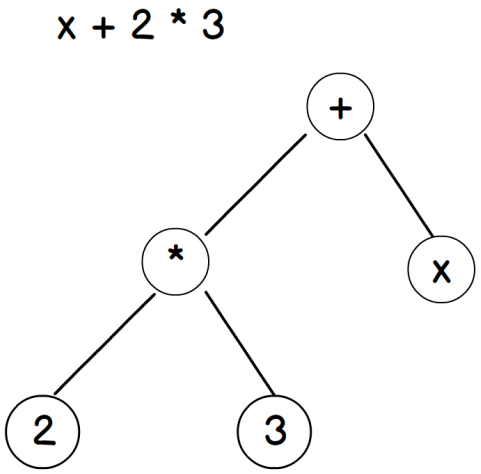
x = 1

x -> x + 2

x,y -> x + y

```
x => x + 2
```





LINQ

GroupBy