

# Семинар 7 по ООП

Ламбда функции, Линеен едносвързан списък



After learning lambdas

# Що е ламбда функция

- Анонимна функция (няма име).
- Удобно е за малки функции, които се викат като аргументи на функции от по-висок ред
- Лесни за описание, връщат указател към функция
- Синтаксис:  $\lambda(\text{аргументи}) \rightarrow \text{тип на връщане} \{ \text{тяло} \}$

# Пример

```
int main()
{
    int arr[] = {1,2,3,4};
    int* arrMultBy2 = map(arr, 4, [](int el)->int{return el * 2;});
    for(int i = 0; i < 4; i++)
    {
        std::cout << arrMultBy2[i] << " ";
    }
}
```

# Std::function

- Указател към функция на стероиди
- Обвивка за указател към функция
- Може да му се присвояват ламбди и адреси към функции
- Синтаксис: `std::function<тип на връщане(аргументи)>`

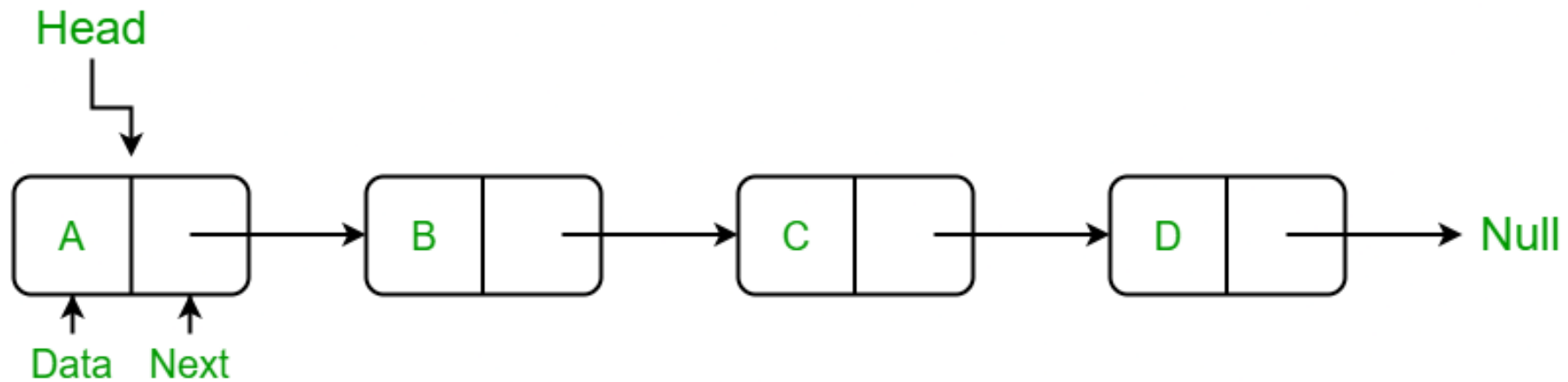
# Пример

```
std::function<int(int)> multBy2 = [](int el)->int {return el * 2;};  
std::cout << multBy2(5) << std::endl;
```

# Линеен едносвързан списък

- Доста безполезна структура (Лично мнение)
- Добре илюстрира обаче как се пишат динамични структури
- Тя представлява кутийки с елементи, разположени на произволно място в паметта, свързани с указател, като всеки има достъп само до следващия.

# Схема



# Задачи

- Напишете реализация на линеен едносвързан списък, която да има функции `add()`, `removeLast()`, `removeFirst()`, `removeOnIndex()`, `head()`, `getAt()`
- Реализирайте функции `filter()` и `map()` за едносвързани списъци, като ги тествате със ламбда.
- Напишете функция за сортиране на линеен едносвързан списък
- Добавете функции `insertAt()`, `removeAt()`, `last()`
- Направете функция `merge`, която слива два сортирани списъка