Deadline: 5 мая 23.59.

В своем репозитории:

1. Перейти на master (git checkout master)
2. Создать ветку (git checkout -b collection)
3. После выполнения (git add .) => (git commit -m “message”) => (git push -u origin collection)
4. Создать пул реквест
5. Отправить ссылку на ваш PR  
     
     
   Реализовать коллекцию, которая будет содержать в себе элементы определенного типа(для примеров выбран тип int)

Требования:

1. Создать отдельный package для коллекции
2. Использовать массивы и слайсы запрещено
3. Коллекция должна быть структурой с методами описанными ниже

Реализовать следующие методы:

Add(element int) – добавляет элемент в коллекцию

Get(index int) – возвращает элемент с данным индексом из коллекции (первый элемент имеет index = 0) (сигнатура: func (collection \*Collection) Get(index int) \*Element)

Remove(index int) – удаляет элемент с данным индексом из коллекции

First() – возвращает первый элемент в коллекции (пример сигнатуры метода:

func (collection \*Collection) First() \*Element  
)

Last() – возвращает последний элемент в коллекции (пример сигнатуры метода:

func (collection \*Collection) Last() \*Element  
)

Length() – возвращает размер коллекции

Print() – выводит коллекцию в консоль (в одну строку)

Реализовать структуру узла коллекции.

Методы:

Next() – возвращает следующий элемент коллекции

Prev() – возвращает предыдущий элемент коллекции

Value() – возвращает значение узла

В main.go:

1. заполнить коллекцию произвольными значениями
2. вызвать каждый метод коллекции (после Add и Remove вызывать Print)

Пример main.go:

func main() {

collection := list.Collection{}

for i := 0; i < 10; i++ {

collection.Add(i)

}

collection.Print()

collection.Remove(4)

collection.Print()

fmt.Println(collection.Get(3))

fmt.Println(collection.First())

fmt.Println(collection.Last())

fmt.Println(collection.Length())

}

