

## **ШИНЖЛЭХ УХААН ТЕХНОЛОГИЙН ИХ СУРГУУЛЬ**

**Мэдээлэл холбооны технологийн сургууль**



**БИЕ ДААЛТЫН АЖЛЫН ТАЙЛАН**

Хичээл : Өгөгдлийн бүтэц ба алгоритм (F.CSM203)

Бие даалтын нэр: Оюутны дунд бүртгэлийн систем

Багш: Д. Батмөнх

Гүйцэтгэсэн оюутан 1 : Н.Мөнхбаяр

Гүйцэтгэсэн оюутан 2 : О.Хүслэн

Оюутны код 1: B242270058

Оюутны код 2: B242270073

# Агуулга

1 Оршил .....	3
2 Системийн бүтэц .....	3
2.1 Үндсэн функцууд.....	3
2.2 Файлын бүтэц .....	3
3 Архитектур болон дизайн .....	4
3.1 UML диаграмм .....	4
3.2 Классын зохион байгуулалт .....	4
3.2.1 Car класс.....	4
3.2.2 CarParking класс .....	4
4 Алгоритм.....	5
4.1 Машин орох алгоритм.....	5
4.2 Машин гарах алгоритм .....	5
5 Хэрэглэх заавар .....	5
5.1 Системийн шаардлага.....	5
5.2 Суулгах заавар .....	5
6.1 Unit Test жишээ.....	6
6.2 Туршилтын үр дүн .....	6
7 Дүгнэлт .....	7
7.1 Хүрсэн үр дүн.....	7
7.2 Өгөгдлийн бүтцийн үр ашиг .....	7
7.3 Ирээдүйн хөгжүүлэлт .....	7

# Зураг

Зураг 1 .....	Error! Bookmark not defined.
---------------	------------------------------

# 1 Оршил

Энэхүү техникийн тайлан нь "Машины зогсоолын систем"-ийн бүтээгдэхүүнийг тайлбарладаг. Систем нь Java программчлалын хэл дээр суурилсан бөгөөд стек өгөгдлийн бүтцийг ашиглан машины зогсоолын үйл ажиллагааг автоматжуулдаг.

Систем нь машинуудыг зөвхөн нэг талаас нь оруулж, гаргах боломжтой зогсоолыг загварчилж, LIFO (Last-In-First-Out) зарчмыг хэрэгжүүлсэн.

<https://github.com/MoonLigth247/Ugugdliin-butets-biydaalt-2>

## 2 Системийн бүтэц

### 2.1 Үндсэн функцууд

Систем нь дараах үндсэн функцуудийг гүйцэтгэдэг:

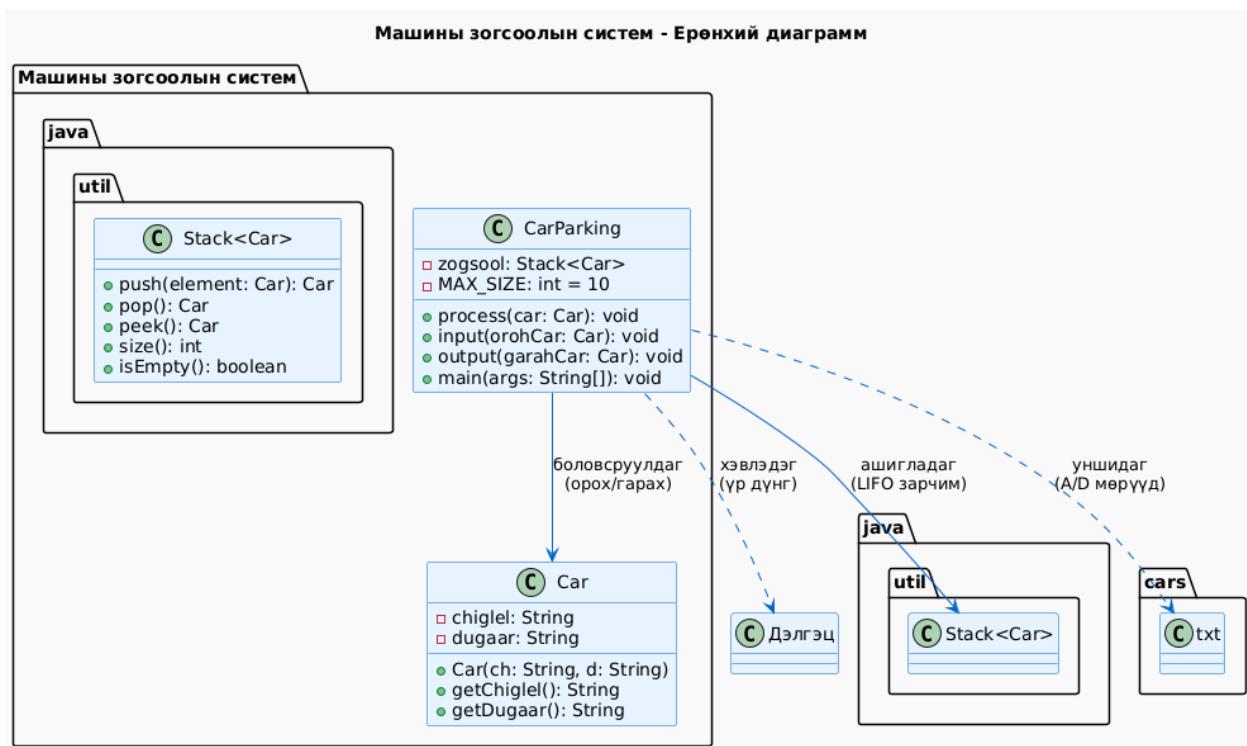
- Машин зогсоолд орох (Arrival)
- Машин зогсоолоос гарах (Departure)
- Зогсоолын чөлөөт зайд шалгах
- Машин гаргах үед бусад машинуудыг түр гаргах
- Гаргасан машинуудыг буцаан зогсоолд оруулах
- Файлаас машины хөдөлгөөний мэдээлэл унших

### 2.2 Файлын бүтэц

- CarParking.java - Үндсэн программ
- Car.java - Машины мэдээллийн класс
- cars.txt - Машины хөдөлгөөний мэдээлэл

# 3 Архитектур болон дизайн

## 3.1 UML диаграмм



Зураг 1

## 3.2 Классын зохион байгуулалт

### 3.2.1 Car класс

Машины үндсэн мэдээллийг хадгалах класс. Машины чиглэл (A - орох, D - гарах) болон дугаарыг хадгална.

### 3.2.2 CarParking класс

Үндсэн системийн логикийг гүйцэтгэдэг класс. Зогсоолын үйл ажиллагааг боловсруулж, файл унших, машины хөдөлгөөнийг зохицуулах зэрэг үйлдлүүдийг гүйцэтгэдэг.

## 4 Алгоритм

### 4.1 Машин орох алгоритм

Хайж буй машин олдох хүртэл дээд машинуудыг түр стек рүү шилжүүлж, хайж буй машин олдвол хэдэн машин шилжүүлснийг тоолно.

### 4.2 Машин гарах алгоритм

Хайж буй машин олдох хүртэл дээд машинуудыг түр стек рүү шилжүүлж, хайж буй машин олдвол хэдэн машин шилжүүлснийг тоолно.

## 5 Хэрэглэх заавар

### 5.1 Системийн шаардлага

- Java JDK 8 эсвэл хожим
- 1GB RAM ба түүнээс дээш
- 100MB чөлөөт дискний зайд

### 5.2 Суулгах заавар

1. Кодыг компайл хийх

```
javac CarParking.java Car.java MyStack.java
```

2. Программ ажиллуулах

```
java CarParking Бичил шалгалт
```

## 6.1 Unit Test жишээ

```
// 2. Машин орох тест

public static void testCarArrival() {

    System.out.println("==> Машин орох тест ==>");

    CarParking parking = new CarParking()

    // Хоосон зогсоолд орох

    String output1 = captureOutput(() -> parking.input(new Car("A", "TEST-01")));

    assertEquals("There is room", output1, "Хоосон зогсоолд орох");

    // Дүүрсэн зогсоолд орох

    for (int i = 2; i <= 10; i++) {

        parking.input(new Car("A", "CAR-" + i));

    }

    String output2 = captureOutput(() -> parking.input(new Car("A", "OVERFLOW")));

    assertEquals("Garage full", output2, "Дүүрсэн зогсоолд орох");

    System.out.println("✓ Машин орох тест амжилттай\n");

}
```

## 6.2 Туршилтын үр дүн

- Бүх unit test амжилттай гүйцэтгэгдсэн
- Машин орох үйлдэл зөв ажилласан
- Машин гаргах үйлдэл зөв ажилласан
- Зогсоол дүүрсэн тохиолдолд алдааны мэдээлэл зөв гарсан
- Файл унших үйлдэл алдаагүй ажилласан

## 7 Дүгнэлт

### 7.1 Хүрсэн үр дүн

Энэхүү "Машины зогсоолын систем" нь дараах үр дүнгүүдийг гаргасан:

- Стек өгөгдлийн бүтцийг практик дээр амжилттай ашигласан
- LIFO зарчмыг машины зогсоолын бодит асуудалд хэрэгжүүлсэн
- Файл боловсруулалт, машины хөдөлгөөний зохицуулалт зэрэг үндсэн алгоритмуудыг хэрэгжүүлсэн
- Объект хандалттай программчлалын зарчмуудыг баримтлан зохион бүтээсэн
- Модульчлогдсон дизайныг хэрэгжүүлсэн

### 7.2 Өгөгдлийн бүтцийн үр ашиг

Үйлдэл	Нарийн төвөгтэй байдал	Тайлбар
Машин орох	O(1)	Стект элемент нэмэх тогтмол хугацаа
Машин гарах	O(n)	Хамгийн муюу тохиолдолд бүх машинуудыг шилжүүлэх
Файл унших	O(n)	n-файлын мөрийн тоо
Чөлөөт зайд шалгах	O(1)	Стекийн хэмжээг шалгах

### 7.3 Ирээдүйн хөгжүүлэлт

Систем нь өргөтгөх боломжтой бөгөөд ирээдүйд дараах шинэчлэлүүдийг хийх боломжтой:

- График хэрэглэгчийн интерфейс нэмэх
- Олон зогсоолтой систем болгон өргөжүүлэх
- Машины цагийн талон, төлбөр тооцооны систем нэмэх
- Веб үйлчилгээ болгон хөгжүүлэх
- Бодит цагийн дэлгэцэд дээр машины байрлалыг харуулах

