

ШИНЖЛЭХ УХААН ТЕХНОЛОГИЙН ИХ СУРГУУЛЬ
МЭДЭЭЛЭЛ, ХОЛБООНЫ ТЕХНОЛОГИЙН СУРГУУЛЬ

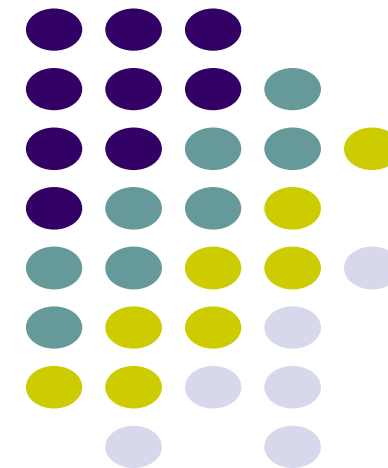
F.CS202
ОБЪЕКТ ХАНДЛАГАТ ПРОГРАМЧЛАЛ

Лекц №15

**Класс болон гишүүдийг тодорхойлох,
классын зохиомж гаргах**

док., дэд проф. Б.Батзолбоо
маг. Б.Мөнхбуян

2021 он



Агуулга



- Классын зохиомж гаргах тухай,
- Нэр-үйл үгийн арга,
- CRC картын арга.

Классын зохиомж гаргах



- Классын зохиомж гаргах гэдэг нь програмд ямар классууд байх болон тэдгээрийн уялдаа холбоо, классууд ямар өгөгдөл, үйлдэлтэй болохыг тодорхойлох үйл ажиллагаа юм.
- Зөв классын зохиомжтой програм нь
 - програмыг ажиллуулахад хялбар, хурдан, алдаагүй байна.
 - програмын алдаа засах, програмд шинэ үйл ажиллагаа нэмэх гэх мэт өөрчлөлтөнд тэсвэртэй байна.

Классын зохиомж гаргах



- Системд классын зохиомж зөв буруу эсэх нь програмын нэвтрүүлэлтийн үед илэрдэг.
- Буруу классын зохиомжтой програм нь алдаа ихтэй, хөрвүүлэх хугацаа удаан, удаан ажилладаг, олон жилийн арчилгаа шаарддаг. Иймээс хэрэглэгч болон зохиогч хоёртоо муу нөлөөтэй байдаг.



10 сая \$ -оос дээш өртөг бүхий систем хөгжүүлэх
төслүүдийн 30% амжилтгүй болдог.



Систем хөгжүүлэх амьдралын цикл

1. Хэрэгцээ шаардлага тодорхойлох,
2. Шинжилгээ хийх,
3. **Зохиомж гаргах,**
4. Хөгжүүлэх,
5. Шалгах,
6. Нэвтрүүлэх,
7. Арчлах.

Классын зохиомж гаргах аргачлал



- Классын зохиомжийг бүхлээр нь гаргадаг аргачлал байхгүй. Ер нь нэг асуудлыг олон янзын классын зохиомжоор шийдэж болно. Нэг бодлогыг олон алгоритмаар бодож болно.
- Туршлагатай програм зохиогч бодлогын даалгавраас классын зохиомж гаргах чадвартай байдаг.



Классын зохиомж гаргах аргачлал

- Гэвч системийн шинжилгээ хийх үед классуудыг илрүүлдэг, уялдаа холбоог нь тодорхойлдог тусдаа аргачлалууд байдаг.
- Классуудыг тодорхойлох хамгийн хялбар арга бол **нэр-үйл үгийн арга** юм.
- Классуудын холбоо хамаарлыг тодорхойлдог хамгийн хялбар арга бол **CRC картын арга** юм.

Нэр-үйл үгийн арга



- Энэ арга нь классуудыг системийн бичиг баримтаас эсвэл бодлогын даалгавраас тодорхойлдог.
- Систем дэх классууд нь системийн бичиг баримт, бодлогын даалгавар дахь нэр үгс, харин үйлдлүүд нь үйл үгс байна. Үйл үгэнд захирагдах тусагдахуун, байц гишүүн нь тухайн үйлдлийн параметр болно.

Нэр үг	Үйл үг
Кино театрын билет захиалгын систем	
	Хадгалах (суудал захиалгууд)
	Хадгалах (утасны дугаар)
Суудал захиалга	
Театр	
	Байх(суудлууд)
Суудал	
Эгнээ	
Үйлчлүүлэгч	
	Урьдчилан захиалах(суудлууд)
	Өгөгдөх(эгнээний дугаар, суудлын дугаар)
	Хүсэлт гаргах(суудал захиалга)
эгнээний дугаар	
суудлын дугаар	
Кино гаргалт	
	Урьдчилан төлөвлөгдсөн(театрт)
Кино	
Огноо	
цаг	
утасны дугаар	

Кино театрын билет захиалгын систем



Кино театрын билет захиалгын систем нь олон кино театрын хувьд суудал захиалгуудыг хадгална. Театр бүр эгнээ бүхий суудлуудтай байна. Үйлчлүүлэгч суудал урьдчилан захиалах бөгөөд захиалгад эгнээний дугаар болон суудлын дугаар өгөгдөнө. Үйлчлүүлэгч хэд хэдэн зэрэгцээ суудал захиалга хийх хүсэлт гаргаж болно.

Захиалга бүр тухайн нэг кино гаргалттай холбоотой байна. (кино гаргалт гэдэг нь өгөгдсөн киног тодорхой цагт дэлгэцэнд гаргана.) Кино гаргалтад огноо, цагийг оноосон байх бөгөөд кино гаргалт нь театрт урьдчилан төлөвлөгдсөн байна. Систем үйлчлүүлэгчийн утасны дугаарыг хадгална.



- Нэр үгс нь системд класс байдлаар оршин байх магадлалтай классуудыг тодорхойлдог. Зарим нэр үг хэрэглэгдэхгүй байж болно. Мөн дараа нь класс нэмэгдэж болно. Зорилго биелүүлэхэд хангалттай мэдээлэл бүхий нэр үгсийг класс болгоно.
- Нэр үгс нь ганц тоон дээр байхад хялбар байдаг. Хэрэв нэр үг нь олон тоотой байвал нэр классаас массив үүсэх боломжтой.



Нэр-үйл үгийн аргын алхамууд

- Бодлогын даалгаврыг өгүүлбэрээр бичнэ.
- Нэр болон үйл үгсийг түүж бичнэ.
- Нэр үгсээс классуудад нэр дэвшүүлнэ.
- Үйл үгсээс үйлдлүүдийг нэр дэвшүүлнэ.
- Класс бүрд төлөв илэрхийлсэн өгөгдөл болон тохирох дутуу харьцаануудыг нэмнэ.
- Орхисон класс болон үйлдлүүдийг нэмнэ.
- Классууд хоорондын харилцааг авч үзэж, ямар класс/үйлдэл бусад ямар класс/үйлдлийг ашиглах/дуудах вэ ? Энэ кодыг бичнэ.
- Үлдсэн дэлгэрэнгүйг бичнэ.

Нэр үг	Үйл үг
Номын сангийн систем	
- оюутнууд	
- номууд	
	Нэвтрэх (Номын санч)
	Ном хайх(barcode)
	Оюутан хайх(оюут.код)
Номын санч	
- Хэрэглэгчийн нэр	
- Нууц үг	
Оюутан	
- Нэр	
- Код	
	Ном олгох(Ном)
	Ном буцаах(Ном)
Ном	
- Нэр	
- Barcode	
	Олгох(3 хоног)
	Сунгах(2 хоног)
	Торгууль бодох(1000 төг)



Жишээ

- **Номын санч номын сангийн системд хэрэглэгчийн нэр, нууц үгээр нэвтэрнэ.** Системд **оюутнууд** болон **номууд** хадгалагдана.
- Номын санч **оюутны нэр, кодыг** үйлчилгээний картаас харж **оюутанд номыг гэрээр олгоно.** Номыг 3 хоногын хугацаатай **олгоно.** Номын хугацааг 2 хоногоор **сунгана.** Номыг хугацаандаа **буцаана.** Хугацаа хэтэрвэл номын хугацаа хэтэрсэн хоног бүрт 1000 төгрөгөөр **торгуул бодно.** Ном бүр **нэр, баркодтой** байна. Системд номыг баркодоор **хайна.** Оюутныг кодоор **хайна**

Жишээ

- Жижиг агуулахаас худалдаа хийдэг нэг програм зохиоё гэж үзье. Програмд зарагдсан барааг хайж олоод бараа материалын бүртгэлээс устгах, шаардлагатай барааг захиалах, татвар бодох болон баримт хэвлэх үйлдлүүдтэй.

```
/** Гүйлгээ хийх үйл ажиллагааг
удирдах класс */
public class Sale {}
/** Үндсэн класс. */
public class Store {}
/** Зарагдах зүйлийг илэрхийлэх
класс */
public class Item {}
/** Агуулах дах бүх бараа. Өөрөөр
хэлбэл бараануудын цуглуулга */
public class Inventory{}
/** Татвартай холбоотой асуудлыг
зохион байгуулах. */
public class Tax {}
/** Үйлчлүүлэгчийн баримттай
ажиллах класс. */
public class Receipt {}
```

```
/* Агуулах дах бүх зүйлсийн жагсаалт. Өөрөөр хэлбэл
бараануудын цуглуулга */
public class Inventory {
    /** Агуулахаас заасан барааг устгах */
    public void remove(Item thing)
    { }
    /** Агуулахад заасан барааг нэмэх */
    public void add(Item thing)
    { }
    ... гэх мэт.
}
/** Татвартай холбоотой асуудлыг удирдана. */
public class Tax {
    public static double calculate(double cost)
    { }
}
/** Үйлчлүүлэгчийн баримтыг удирдах класс */
public class Receipt {
    public void print()
    { }}
```





```
public class Item {  
    private String name = null;  
    private String description = null;  
    private double cost = 0.0;  
    public Item(String name, String  
description, double cost)  
    {...}  
    public double getCost()  
    { ... }  
...etc.  
}
```

```
public class Receipt {  
    Item[] purchasedItems = null;  
    double tax = 0.0;  
    public Receipt(Item[] purchasedItems)  
    {...}  
    public void print()  
    {  
        ...  
        for(int i=0; i<purchasedItems.length; i++)  
        {  
            totalCost += Item[i].getCost();  
        }  
        double tax = Tax.calculate(totalCost);  
        ...  
    }  
}
```

CRC карт ашиглах

- Нэр-үйл үгийн арга нь классуудыг тодорхойлдог.
- Харин энэ арга нь классуудын хоорондын харилцан үйлчлэлийг тодорхойлдог.
- Ингэхдээ системд хамтарч ажиллаж буй хүмүүстэй хэлэлцэж шийдвэрлэдэг.



CRC карт



- CRC гэдэг нь Class/Responsibilities/Collaborators буюу класс/үүрэг хариуцлагууд/хамтран ажиллагч гэсэн үг.
- Гол санаа нь класс бүрд нэг карт ашиглана. Үүнд бодитой картууд ашиглаж, хамтран хэлэлцэх замаар бөглөнө. Карт бүр 3 хэсгээс бүрдэнэ.

Классын нэр	Бусад класстай
Үүрэг хариуцлагууд	харилцан үйлчлэлүүд



Анхаарах зүйлс

- CRC карт үүсэх үед програмын бүтцийг ойлгох боломжтой болсон байна. Классууд нь хамтарч програмын зорилгыг биелүүлнэ.
- CRC картуудаас зарим нь хэрэглэгдэхгүй байвал түүнийг орхино. Картууд нь шаардлагатай классуудын олонлогийг үзүүлнэ.
- Класс бүрийн хандалтыг тодорхойлох шаардлагатай.
- Гишүүн бүр public харагдалттай байж болохгүй.

CRC аргын алхамууд



- Картууд аваад,
- Карт бүрт
 - Нэг классын нэр бичнэ.
 - Классын үүрэг хариуцлагуудыг бичнэ,
 - үйлдлүүд/объектод байх хувьсагчид
 - Хамтран ажиллагч классуудыг бичнэ.
 - Түүнийг объектын хувьсагчид байдлаар ашиглах классууд.
 - Байгуулагч болон үйлдлүүд рүү хувьсагч байдлаар дамжуулах классууд.
- Хялбар болгохын тулд карт тус бүрийг олон дахин авч үзэж орхисон үүрэг хариуцлага болон классыг нэмнэ.
- Бодит классууд руу буулгах.



Класс: Бараа материал

Үүрэг хариуцлагууд:

- Дэлгүүрийн бараа бүрийн объектыг хадгалах
- Бараа устгах
- Бараа нэмэх

Хамтран ажиллагч:

- Бараа класс (Бараа хадгалахад)
- Давтагч класс (Бараа устгах нэмэх үйлдлийг хэрэгжүүлэхэд)

- Жиш: хэрэв карт хэтэрхий олон үүрэг хариуцлагатай бол түүнийг дэд классуудад хуваана.

Хүндрэлүүд



- Класс нь аль болох цөөн үйлдэл гүйцэтгэдэг байх ёстой.
- Классууд объектын ерөнхий дүрслэлийг илэрхийлнэ.
- Объектууд бие даасан үйлдлүүдтэй байна.
- Хэтэрхий олон үйлдэл хийдэг класс нь бие даасан үйлдлүүдтэй байж чаддаггүй.
 - Классууд нь дэд классуудад задарч болно.
 - Машин класс дугуй, хөдөлгүүр, шил зэрэг тохиолдлын хувьсагчидтай байна. Өөрөөр хэлбэл энгийн классуудаас бүрдэнэ.

ДҮГНЭЛТ



- Классын зохиомж нь програмын ажиллагаа, чанарт чухал нөлөөтэй.
- Нэр-үйл үгийн арга нь класс тодорхойлоход, харин CRC картын арга нь классуудын хоорондын хамаарлыг тодорхойлоход тус тус ашиглагдана.