# Классийн удамшил ба байгуулагч функц (Лаборатори №8)

Ү.Мэндсайхан

МУИС, ХШУИС, Компьютерын ухаан, mendsaikhan@gurvanshiidel.mn

# 1. ОРШИЛ

Энэхүү лаборитароор this хувьсагч гэж юу болох, түүнийг комплиатор хэрхэн хэрэглэгдэг талаар, мөн static гишүүн өгөгдөл, static гишүүн функцийн талаарх ойлголтуудыг судалж, хэрэгжүүлсэн.

# 2. ЗОРИЛГО

Классд түүнээс үүссэн бүх объектийг тоолдог статик хувьсагч нэмж үүссэн объектуудыг тоолох болон өөр хоорондоо ялгаатай дүрсүүдийг талбай болон периметрээр нь эрэмблэх дасгал гүйцэтгэнэ. Үүний тулд дараах зорилтуудыг тавьж ажилласан:  
 2.1 Гишүүн функцээс гишүүн функц рүү хандахдаа this хувьсагч ашиглах  
 2.2 Статик гишүүн функц болон статик гишүүн өгөгдлийг тодорхойлох

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 This хувьсагч гэж юу вэ?, үүнийг комплиатор хэрхэн хэрэглэгдэг вэ?

This хувьсагч нь зөвхөн классын гишүүн функц дотор ашиглагдана. Комплятор гишүүн функцийг дуудахаас өмнө функц дотор хамгийн эхэнд тухайн классын this гэх объектон хаяган хувьсагч зарлаж өгдөг. This хувьсагчийг ашигласнаар программ нь уншууртай ойлгоход хялбар болж өгдөг.

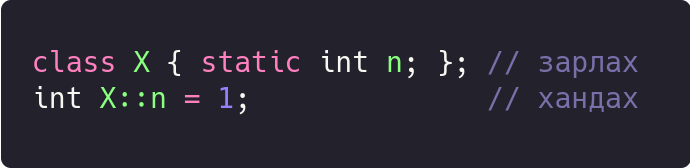
## 3.2 Static гишүүн өгөгдөл гэж юу вэ? Хэрхэн зарладаг вэ?

Статик гишүүн өгөгдөл нь ямар ч объекттой холбогдоогүй, классын гишүүн өгөгдлийг хэлнэ. Статик гишүүн өгөгдлийг объект дунд ерөнхий байдлаар үүсгэдэг. Статик гишүүн өгөгдлийг  
классын тодорхойлолт дотор статик гэх түлхүүр үг ашиглан зарладаг. Гишүүн гишүүд руу объектоор эсвэл классаар нь дамжин хандаж болно. Статик хувьсагч нь ихэнхдээ тогтмол хувьсагч зарлахад ашиглагддаг.

## 3.3 Static гишүүн өгөгдлийн амьдралын мөчлөг ямар байдаг вэ?

Статик гишүүн өгөгдлийн амьдралын мөчлөг нь программын амьдралын мөчлөгтэй адил гэж хэлж болно. Програм эхлэх үед үүсэж програм дуусах үед устдаг.

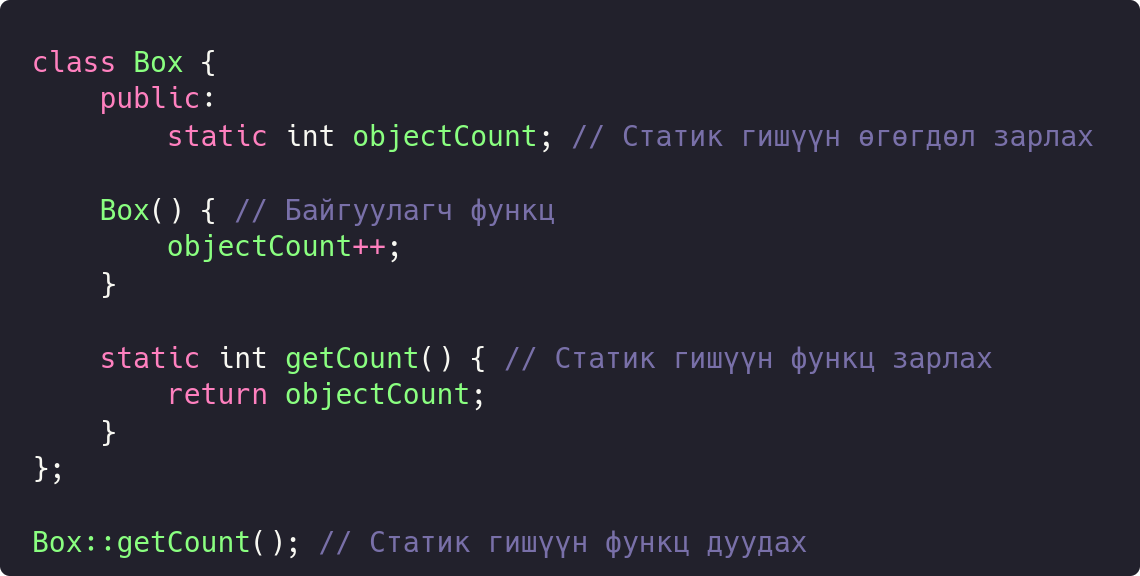
## 3.4 Static гишүүн өгөгдөлд хандах

Статик гишүүн өгөгдөлд көвхөн классаар нь дамжуулж хандана. Ингэхдээ классын тодорхойлолтын гадна тухайн статик гишүүн өгөгдлийн төрөл классын нэр үйлчлэх хүрээний оператор статик гишүүн өгөгдөл гэх маягаар хандана.  


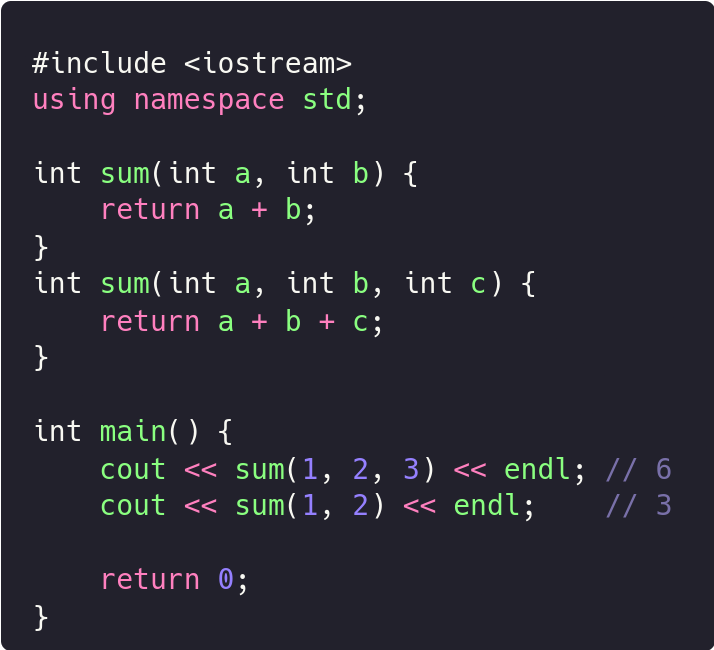
## 3.5 Static гишүүн өгөгдөл ямар ашигтай вэ?

Статик гишүүн функц глобал үйлдэл гүйцэтгэх эсвэл классын бүх глобал өгөгдөлд хэрэглэдэг. Нэг удаа л зарлагддаг болхоор санах ойд сайн нөлөөтэй.

## 3.6 Static гишүүн функцийг хэрхэн зарладаг, дууддаг вэ?

Статик функцийг классын тодорхойлолт дотор зарлахдаа static тусгай үгийг ганц удаа хэрэглэнэ. Түүнийг функцийн тодорхойлолтын толгой хэсэгт огт хэрэглэхгүй.  


## 3.7 Функц дахин тодорхойлох гэж юу вэ?

Ижил нэр, төрөл нь ижил боловч параметрийн тоо, төрөл, дараалал өөр байдаг олон функц тодорхойлох үйлийг функц дахин тодорхойлох гэнэ.  


## 3.8 Жинхэнэ хийсвэр функц гэж юу вэ?

Зөвхөн зарласан, тодорхойлогдоогүй функцийг жинхэнэ хийсвэр функц гэнэ. Мөн жинхэнэ хийсвэр функц нь хүүхэд класстаа дахин тодорхойлогдсон байх ёстой.

## 3.9 Хийсвэр класс гэж юу вэ? Хэрхэн ашигладаг вэ??

Хийсвэр класс гэдэг нь тухайн классаас шууд обьект үүсгэн ашиглах боломжгүй, зөвхөн удамшуулан хэрэглэх зориулалттай класс юм.

# 4. ХЭРЭГЖҮҮЛЭЛТ 4.1Shape классын байгуулагч функц дуудагдах бүрт статик тоолох хувьсагчийг нэгээр нэмэгдүүлэх зарчмаар үүссэн объектийг тоолсон.

# 4.2 Өмнө зарласан static хувьсагчид утга оноох, утгыг нь авах static гишүүн функц бичих

# 4.3 Классын гишүүн функцууд дотор гишүүн функц руу хандахдаа өмнө нь this хувьсагч ашиглах

# 4.4 Хоёр хэмжээст дүрс классд приметер олох, талбай олох функцуудыг жинхэнэ хийсвэрээр зарлаж хүүхэд классуудад дахин тодорхойлно

# 4.5 Өөр хоорондоо ялгаатай дүрсүүдийг талбай болон приметерээр нь эрэмбэлнэ. Ингэхдээ приметер болон талбай олох жинхэнэ хийсвэр функцуудыг ашиглана.

5. ДҮГНЭЛТ

Хийсвэр класс ашигланаар объект үүсэхгүй. Хэрэв объект үүсгэх бол зөвхөн хүүхэд классаар нь дамжуулж үүсгэж болно. This хувьсагч ашиглснаар програм уншууртай болж өгдөг. Хийсвэр классыг сайн хэрэгжүүлснээр ерөнхийлөл хийхэд үр ашигтай.

# 6. АШИГЛАСАН МАТЕРИАЛ

1. Объект хандлагат технологийн С++ програмчлал, Ж.Пүрэв, 2008, Улаанбаатар

# 7. ХАВСРАЛТ

Github линк, [Lab 8](https://github.com/mendeeislife/object/blob/main/lab8/lab8.cpp)