(Лаборатори №7)

Э.Багабанди

ХШУИС - ийн ПХ хөтөлбөрийн

3-р түвшний оюутан, [19B1NUM0700@num.edu.mn](mailto:19B1NUM0700@num.edu.mn)

1. **ОРШИЛ/УДИРТГАЛ**

Уг лабораторийн хүрээнд өгөгдсөн дасгалуудыг шийдвэрлэж, тайлбарлахад ашиглагдсан C++ хэлний онолын ойлголт /удамшил/ - уудыг судалж, эзэмшсэн болно.

2. **ЗОРИЛГО**

Энэхүү лабораторийн ажлын хүрээнд С++ хэлний удамшил, түүний горим/public, private, protected/, давуу тал болон төрлүүдийн талаар судалж, үүнийг ашиглан лабораторт өгөгдсөн дасгалуудыг гүйцэтгэн жижиг хэмжээний хэрэгжүүлэлт хийх юм.

3. **ОНОЛЫН СУДАЛГАА**

3.1. Удамшил гэж юу вэ? (тодорхойлолт, C++ дээр хэрхэн хэрэгжүүлдэг талаар

бичнэ)

* Удамшил гэдэг нь нэгэнт бий болсон зүйлийг дахин хэрэглэхтэй холбоотой ойлголт бөгөөд туршиж бэлэн болгосон классын кодыг өөр системд дахин хэрэглэх буюу эргэн хэрэглэгдэх чадвар, чадамжтай болгодог. Өөрөөр хэлбэл бэлэн байгаа классаас шинэ класс үүсгэх арга технологи гэж хэлж болох бөгөөд энэ үед шинээр үүсч буй класс нь өмнөх классынхаа шинжийг өвлөн авна.
* Өөрөөс нь класс үүсч байгаа классыг үндсэн(дээд/эх) класс, шинээр үүсч байгааг нь удамших, удамшсан(дэд/охин/хүүхэд) класс гэнэ.
* Удамшиж үүсэх класс нь эх классынхаа шинж, аргаас өвлөж авахаас гадна өөрийн гэсэн шинжтэй, өөрөөр хэлбэл эх класст байхгүй гишүүдтэй байж болно. Эх классын гишүүн функцийг удамших класс дотор дахин тодорхойлж болох ба ингэхэд эх классын эх код заавал байх албагүй юм.

3.2. Удамшлын горим. public, private, protected горимын талаар тайлбарлаж

жишээгээр батална.

* **Классыг public горимоор удамшуулж үүсгэх** үеийн удамших классын тодорхойлолт:



* Эх класс public горимоор удамших тохиолдолд түүний public гишүүд удамшиж үүсэх классын public гишүүд болно. Иймээс удамших классын функц ба объект эх классынхаа public гишүүд рүү хандана. Харин эх класс public горимоор удамших үед түүний private гишүүд удамших хэдий ч удамших классын гишүүд эх классынхаа private гишүүд рүү хандаж чадахгүй. Мөн эх класс public горимоор удамших үед түүний protected гишүүд удамших классынхаа protected гишүүд болох тул уг гишүүд рүү удамших классын функцээс хандаж болно.
* Тухайлбал,



* **Классыг private горимоор удамшуулж үүсгэх** үеийн удамших классын тодорхойлолт:



* Эх класс private горимоор удамших тохиолдолд түүний public болон protected гишүүд удамшиж үүсэх классын private гишүүд болно. Иймд удамших классын объект эх классынхаа гишүүд рүү хандаж чадахгүй. Харин функц нь классынхаа private гишүүд рүү хандаж болдог тул эх классынхаа public, protected гишүүд рүү удамших классын функцээс хандана.
* Тухайлбал,

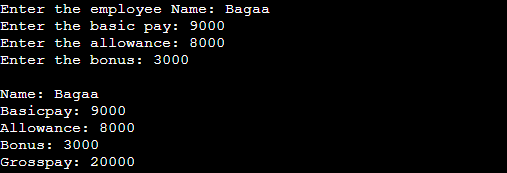


* **Классыг protected горимоор удамшуулж үүсгэх** үеийн удамших классын тодорхойлолт:



* Эх класс protected горимоор удамших тохиолдолд түүний public болон protected гишүүд удамшиж үүсэх классын protected гишүүд болно. Иймд уг гишүүд рүү удамших классын функцээс хандаж болно. Харин эх класс protected горимоор удамших үед түүний private гишүүд удамших хэдий ч удамших классын гишүүд эх классынхаа private гишүүд рүү хандаж чадахгүй.
* Жишээ код:

[[1]](#footnote-1)



* Удамших горимын хүснэгт:

|  |  |  |  |
| --- | --- | --- | --- |
| **Эх классын гишүүд** | **Удамших горим** | | |
| public | private | protected |
| public | public | private | protected |
| private | X | X | X |
| protected | protected | private | protected |

3.3. Удамшлын давуу талуудыг тоочин бичиж бодит жишээн дээр тайлбарла.

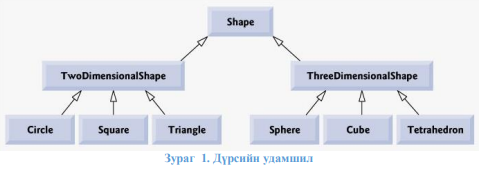
* Програмчлалд удамших шинжийг хэрэглэснээр дараах боломжууд бүрдэнэ.
  + Код болон програмыг **дахин хэрэглэх** – Жишээ код №1 дээрх Employee классыг удамшуулан Engineer классыг үүсгэсэн бөгөөд ингэснээр Employee классын кодыг дахин бичихгүйгээр шууд Engineer класс дотор хэрэглэх боломжтой болсон.
  + Код болон програмыг кодыг **дундаа хэрэглэх** – Жишээ код №1 дээрх Employee классын гишүүн өгөгдөл болон функцийг Engineer класс хандах эрхээс хамааран дундаа хэрэглэх боломжтой байгаа нь харагдаж байна.
  + Код болон програмыг **тогтмол интерфейс хэрэглэх** - Интерфейс буюу гишүүн функцийг цуглуулга бүхий классыг үүсгэн удамшуулж, дэд классуудад дахин тодорхойлон ашиглах боломжтой. Жишээ код №1 дээр Employee классын гишүүн функцийг Engineer класс дахин тодорхойлж ашигласантай ижил.

3.4. Удамшлын хэдэн төрөл байдаг вэ? Тус бүрийг тайлбарлан бич.

* Удамшлын 5 төрөл байдаг. Үүнд:
  + - Нэг – нэг буюу дан, энгийн удамшил. Удамших класс зөвхөн нэг эх класстай байна. Тухайлбал, үхэр нь амьтны аймагт багтах тул cow класс нь animals классаас удамшиж үүссэн гэж үзэж болно.
    - Олон – нэг буюу нийлмэл удамшил. Зарим тохиолдолд удамших буюу дэд класс нь 2 – оос цөөнгүй эх класстай байж болно. Тухайлбал, ачаа ачдаг, мөн олон 4 буюу түүнээс дээш тооны суудалтай машинуудыг суудлын машин болон ачааны машинаас удамшиж үүссэн гэж үзэж болно.
    - Олон түвшинт удамшил. Удамших класс өөрөө удамших классаас үүснэ. Тухайлбал, Z класс Y классаас, Y класс нь харин X классаас үүсэх бол энэ шинжийг олон түвшинт удамшил гэнэ.
    - Шаталсан удамшил. Нэг эх классаас 2-оос цөөнгүй удамших класс үүсэх бол ийм удамшлыг шаталсан удамшил гэнэ. Тухайлбал, МУИС – аас салбар сургуулиуд буюу Ахисан түвшний сургууль, Бизнесийн сургууль, Олон улсын харилцаа, нийтийн удирдлагын сургууль, Хууль зүйн сургууль, Хэрэглээний шинжлэх ухаан, инженерчлэлийн сургууль, Шинжлэх ухааны сургууль, Завхан сургууль, Эрдэнэт сургууль, Байгаль-Эх Лицей Ахлах Сургууль, Экологийн Боловсролын Төв эдгээр нь шаталсан удамшлаар удамшсан гэж үзэж болох юм.
    - Холимог удамшил. Олон – нэг буюу нийлмэл удамшил, шаталсан удамшил хоёрын нийлэмж удамшил юм. Тухайлбал, эвэр туурайтны аймагт багтах гүү, илжиг хоёрын дундаас гарах луус бол уг удамшлаар үүссэн гэж үзэж болно.

4. **ХЭРЭГЖҮҮЛЭЛТ**

Зураг 1-д дүрсэлсэн удамшлын модны дагуу хоёр хэмжээст геометрийн дүрсүүдийн классыг байгуул.

**Тойрог:**

* Гишүүн өгөгдөл-тойргийн төвийн координат, радиус, нэр;
* Гишүүн функц-талбай олох, тойргийн урт тооцох
* Тойргийн төвийн цэг, радиусд утга онооно.



**Квадрат:**

* Гишүүн өгөгдөл-оройнуудын координат, талын урт, нэр;
* Гишүүн функц-талбай олох, приметр тооцох
* Квадратын талын урт, зүүн дээд оройн координатад утга оноогоод бусад оройг тооцоолж классын харгалзах гишүүн өгөгдөлд хадгална.



**Зөв гурвалжин:**

* Гишүүн өгөгдөл-оройнуудын координат, талуудын урт, нэр;
* Гишүүн функц-талбай, приметр тооцох
* Зөв гурвалжингийн талын урт, дээд оройн координатыг утга оноогоод бусад оройг тооцоолно



Классуудад байгаа нийтлэг шинжийг эх классад оруулаад түүнээсээ удамшуулж классыг үүсгэнэ.

5. **ДҮГНЭЛТ**

Энэхүү лабораторын ажлын хүрээнд C++ хэл дээр жижиг хэмжээний програм хөгжүүлсэн бөгөөд тухайн програмыг хэрэгжүүлэхдээ олон түвшинт болон шаталсан төрлийн удамшлыг үүсгэж, public горимоор удамшуулж, эх классуудыг гишүүн өгөгдлийг protected – оор тодорхойлсон. Үүнээс удамших классууд эх классуудынхаа public болон protected гишүүн өгөгдөл болон дүрмүүдэд функц дотроос хандах боломжтой болох юм. Мөн классуудыг удамшуулан үүсгэхийн өмнө, дан дангаар үүсгэн класс бүрт хадгалагдаж байгаа нийтлэг шинж, дүрмүүдийг авч эх классуудыг үүсгэсэн болно. Ингэснээр удамшлын давуу талыг бий болж, дахин хэрэглэгдэх өндөр чадвар, чадамжтай класс үүсгэх боломж бүрдэх юм.

6. **АШИГЛАСАН МАТЕРИАЛ**

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

7. **ХАВСРАЛ**

1. Жишээ код №01 [↑](#footnote-ref-1)