# Хаяган ба Заалтан хувьсагийн тухай ойлголт ба хэрэглээ (Лаборатори №2)

Д.Самдан

ХШУИС, ПХ-ийн 2 түвшин, pri.samdan@gmail.com

# 1. ОРШИЛ

Хаяган хувьсагч болон заалтан хувьсагчийн талаар судалж, тэдгээрийг хэрхэн ашиглах талаар суралцаж авна. Өмнөх лабораторийн ажлаар хэрэглэгчийн функц рүү утгаар нь аргумент дамжуулж сурсан бол энэ лабораториор шинээр сурч авсан заагч болон заалтыг ашиглан функц рүү хэрхэн хаягаар, заалтан хувьсагчаар аргумент дамжуулахыг сурна. Мөн хаяган хувьсагч ба new оператор ашиглан санах ой нөөцөлж авах боломжтой болсон гэвч зөв хэрэглэхгүй бол санах ойн цоорхой үүсгэх аюултай. Сүүлд нь дотоод функц гэж юу болохыг тайлбарлана.

# 2. ЗОРИЛГО

Бид өмнөх C хэлний хичээл дээр энгийн төрөлийн хувьсагчууд буву бүхэл тоо, бодит тоо, тэмдэгт илүү судаж байсан болон энэ лабораторийн ажлаар хаяган болон заалтан хувьсагчуудын талаар судлаж, ашиглаж сурах.

1. Хаяган хувьсагийн тухай судлаж тодорхойлох
2. Заалтан хувьсагийн тухай судлаж тодорхойлох
3. Хаяган хувьсагчийг ашиглан хэрхэн санах ой нөөцөлхийг тухай
   1. Ойн операторуудыг судах
      1. new болон delete операторуудын үүрэг ба хэрэглээ
   2. Санах ойн цоорхой нь талаах судлах
4. Классыг хэрэглэсэн функц тодорхойлох.

Дотоод функцийг талаар судлаж бичнэ.

inline функцийн давуу болон сүл талаад сурах.

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 Класс ба объект

Бодит объектыг програмын орчинд хийсвэрлэхдээ Класс /Class/ гэсэн ойлголтыг ашигладаг [1]. Объект гэдэг нь ...... . Объектын шинжийг классын гишүүн өгөгдлөөр төлөөлүүлдэг ........

Гишүүн өгөгдөл гэдэг нь ............

Гишүүн функц нь .........................

## 3.2 Объектон хүснэгт

Объект нь үүсмэл өгөгдлийн төрөл учир хүснэгт үүсгэж болно. ...........................

## 3.3 Объектон хүснэгт эрэмбэлэх

Эрэмбэлэхэд шаардлагатай хамгийн чухал оператор нь харьцуулах оператор. C++ ийн стандарт жиших операторууд нь /==, >, </ бүхэл объектыг жишиж чадахгүй учир хоёр объектыг жиших функц бичнэ......................

Эрэмбэлэх алгоритм нь .....................

## 3.4 Тэмдэгтийн цуваа харьцуулах функц

Хоёр тэмдэгтийн цувааг харьцуулахад байт байтаар харьцуулдаг. Яагаад гэвэл ..............

Үүнийг хийдэг strcmp гэдэг функц байдаг [2]. Функцийн буцаах утга нь ....., авах параметерүүд нь ............ байна.

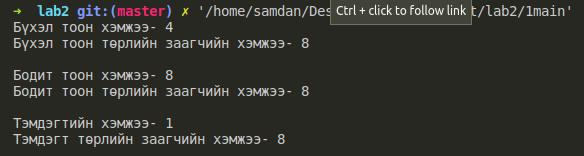
## 3.5 Гэх мэт дэд гарчиг өгч цааш бичнэ

# 4. ХЭРЭГЖҮҮЛЭЛТ

**1-р асуулт:**

Хаяган хувьсагчийн хэмжээг Ubuntu 18.04.1

а гэсэн бүхэл тоо, b гэсэн бодит тоо, c гэсэн тэмдэг гэсэн гурван хаяган хувьсагчуудыг зарлаж өгсөн. Тэдгээрийн санах ойд эзлэх хэмжээгийн sizeof() гэсэн фүнкцийг ашиглан санах ойд эзлэх хэмжээг олж cout ашиглан хэвэлсэн. Үүнээс үзхэд бүгэд адилхан 8byte байна.



# 5. ДҮГНЭЛТ

Өмнө бичсэн зүйлсээ дүгнэнэ. Жишээ нь дахин ашиглах зарчмыг ашигласнаар ... мөр код хэмнэсэн. Классыг загварчилснаар ийм давуу тал гарч байна. Байгуулагч функы ашилавал ийм давуу талтай гэх мэт.

# 6. АШИГЛАСАН МАТЕРИАЛ

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

2. strcmp, <http://en.cppreference.com/w/c/string/byte/strcmp>

# 7. ХАВСРАЛТ

**1-р асуулт**

#include <iostream>;

using *namespace* std;

*int* main()

{

    // Бүхэл тоон заагч

*int* \*a;

    // Бодит тоон заагч

*float* \*b;

    // Тэмдэгт төрлийн заагч

*char* \*c;

    cout << "Бүхэл тоон хэмжээ- " << sizeof(*int*) << endl;

    cout << "Бүхэл тоон төрлийн заагчийн хэмжээ- " << sizeof(a) << endl

         << endl;

    cout << "Бодит тоон хэмжээ- " << sizeof(*double*) << endl;

    cout << "Бодит тоон төрлийн заагчийн хэмжээ- " << sizeof(b) << endl

         << endl;

    cout << "Тэмдэгтийн хэмжээ- " << sizeof(*char*) << endl;

    cout << "Тэмдэгт төрлийн заагчийн хэмжээ- " << sizeof(c) << endl

         << endl;

}

**2-р асуулт**

#include <iostream>;

using *namespace* std;

*int* main()

{

    // Тэмдэгт тоон заагч зарласан

*char* \*p1;

    // Бүхэл тоон заагч зарласан

*int* \*p2;

    // давхаp нарийвчлалтай бодит тоон заагч зарласан

*double* \*p3;

    // p1, p2, p3 - ийн хэмжээг хэвлэж байна.

    cout << sizeof(p1) << " " << sizeof(p2) << " " << sizeof(p3);

    // OS-ийн 32bit / 64 bit ээс хамааран 4 / 8 байна.36+

}

**3-р асуулт**

#include <iostream>;

using *namespace* std;

*int* main()

{

    // а гэсэн бүхэл тоо зарлаж, түүнд 125 гэсэн анхны утга оноож өгнө

*int* a = 125;

    // p гэсэн бүхэл тоон хаяган хувьсагч зарлаж, а-н хаягийг p-д оноож өгнө

*int* \*p = &a;

    // p буюу а-н хаягийг хэлнэ

    cout << p << endl;

    // р-д байгаа хаягийн утга буюу а-125 -г хэвлэнэ

    cout << \*p << endl;

    // p хувьсагч а-н хаяг дээр 1-г нэмэх ба энэ заагч нь бүхэл тоон төрөл

    // учир &a + 4 байна.

    p++;

    // р-н шинэ хаягийг хэвлэнэ

    cout << p << endl;

    // шинэ хаягт байгаа утгыг хэвлэнэ

    cout << \*p << endl;

}

**4-р асуулт**

#include <iostream>;

using *namespace* std;

*int* main()

{

    // Бүхэл тоон 5 элементтэй хүснэгт зарлана

*int* numbers[5];

    // Бүхэл тоон хаяган хувьсагч зарлана

*int* \*p;

    // р-д бүхэл тоон хүснэгтийн анхны элементийн хаягийг өгнө

    p = numbers;

    // р-н анхны утга 10 болж байна.

    \*p = 10;

    // р-н утга буюу number-н эхний элементийн хаягийг 1-ээр нэмж дараагийн элементийн хаягийг авна

    p++;

    // number хүснэгтийн 2-р элементэд 20 гэсэн утга онооно

    \*p = 20;

    // р-д number-н 3-р элементийн хаягийг өгнө

    p = &numbers[2];

    // number-н 3-р элементийг 30 болгоно

    \*p = 30;

    // р-д number хүснэгтийн 4-р элементийн утгыг оноож өгнө

    p = numbers + 3;

    // number-н 4 дахь элементийг 40 болгоно

    \*p = 40;

    // р-д number хүснэгтийн анхны элементийн хаягийг өгнө

    p = numbers;

    // р-д эхний хаяг дээр sizeof(int) \* 4-р нэмж number хүснэгтийн

    // 5 дахь элементийн утгыг 50 болгоно

    \*(p + 4) = 50;

    // number-н эхний 5 элементийг хэвлэж харуулна

    for (*int* n = 0; n < 5; n++)

        cout << numbers[n] << ", ";

}

**5-р асуулт**

#include <iostream>;

using *namespace* std;

/\*

    2 тооны утгуудыг хаяган хувьсагч ашиглан солино

    Params:

    int a - утгыг нь солих тоо

    int b - утгыг нь солих тоо

    Return:

    void

\*/

*void* swap\_nums\_with\_ponter(*int* \*a, *int* \*b)

{

    // temp-т а-н утгыг хадгална

*int* temp = \*a;

    // а-т b-н утгыг оноож өгнө

    \*a = \*b;

    // b-m хадгалж авсан а-н утгыг оноож өгнө

    \*b = temp;

}

*int* main()

{

    // утгыг нь солих 2 бүхэл тоо зарлах

*int* a = 10;

*int* b = 96;

    // функцээ дуудан ажиллуулах

    swap\_nums\_with\_ponter(&a, &b);

    // хоёр тоогоо хэвлэнэ

    cout << "a: " << a << endl;

    cout << "b: " << b << endl;

}

**6-р асуулт**

#include <iostream>

using *namespace* std;

/\*

    2 тооны утгуудыг заалтан хувьсагч ашиглан солино

    Params:

    int a - утгыг нь солих тоо

    int b - утгыг нь солих тоо

    Return:

    void

\*/

*void* swap\_nums\_with\_refrence(*int* &a, *int* &b)

{

    // temp-т а-н утгыг хадгална

*int* temp = a;

    // а-т b-н утгыг оноож өгнө

    a = b;

    // b-m хадгалж авсан а-н утгыг оноож өгнө

    b = temp;

}

*int* main()

{

    // утгыг нь солих 2 бүхэл тоо зарлах

*int* a = 10;

*int* b = 96;

    // функцээ дуудан ажиллуулах

    swap\_nums\_with\_refrence(a, b);

    // хоёр тоогоо хэвлэнэ

    cout << "a: " << a << endl;

    cout << "b: " << b << endl;

}

**9-р асуулт**

#include <iostream>

using *namespace* std;

*int* main()

{

    // Хаяган хувьсагч зарлах

*int* \*p\_a;

    // 2 удаа давтах for давталт үүсгэх

    for (*int* i = 0; i < 2; i++)

    {

        // а гэсэн бүхэл тоон хаяган хувьсагч зарлаж түүнд new int-с нөөцөлсөн хаягийг оноож өгнө

*int* \*a = new *int*;

        // p\_a-т а-н хаягийг өгнө

        p\_a = a;

        // а-н хаяган дээр байгаа утгыг i болгоно

        \*a = i;

        // а-н хаяган дээр байгаа утгыг хэвлэнэ

        cout << "\*a = " << \*a << endl;

    }

    // p\_a-н хаяган дээр байгаа утгыг хэвлэнэ

    cout << "p\_a = " << \*p\_a << endl;

    // p\_a дээр нөөцөлсөн санах ойг устгана

    delete p\_a;

    return 0;

}