# Заалтан хувьсагч гэж юу вэ? Жишээгээр тайлбарла

Заалтан хувьсагч болгон зарлах үед энэ нь зааж буй хувьсагчийн өөр нэмэлт нэр болно. Өөрөөр хэлбэл заалтан хувьсагчийг заасан хувьсагчтай адил эрхтэй хэмээн үзнэ. Энэхүү хувьсагч зарлахад зааавал утга оноох ёстой бөгөөд Заагч шиг чөлөөтөй ашиглаж болохгуй, яагаад гэвэл нэр нь нэг ойн блоктой уягддаг.

Хувьсагчийг зарлахдаа '&' гэж өмнө нь тэмдэглэнэ.

int main()

{

    int value = 5; // жирийн integer

    int &ref = value; // хувьсагчийнутга заана

    value = 6; // value 6 болло

    ref = 7; // value 7 болло

    std::cout << value; // 7 хэвллэ

    ++ref;

    std::cout << value; // 8 хэвэллэ

cout << &value; // 0012FF7C хэвэллэ

cout << &ref; // 0012FF7C хэвэллэ

 return 0;

}

# Хоёр хувьсагчийн байрыг солих функцыг заалтан хувьсагч ашиглан бич

Example: “1.cpp”

# Заалтан болон хаяган хувьсагчийн ялгаа, тус бүрийн давуу талыг тайлбарла

* Заагчийг ахин ашиглаж болно(өөр хувьсагчийн хаягийг заах), заалтан хувьсагч зарлагдахдаа авсан утгаас өөр утга авахгуй.
* Заагч өөрийн гэсэн хаягтай санас ой дээх, заалтан хувьсагч болхоор холбогдсон хувьсагчийн санах ойн хаягийг хатран хэрэглэнэ, ба стэк дээр зай эзлэнэ.
* Заагч заагч руу зааж илүү их давхрага үүсгэдэг, заалт зөвхөн нэг давхарыг үүсгэнэ.
* Заагчинд NULL утга шууд өгч болно, заалтанд болохгуй.
* Заагч хүснэгтээр явж чадна, ++ ашиглаж.
* Заагч \* ашиглаж ойг олж чаддаг, заалт шууд хандана.
* Заагч хаяг хадгалана, заалт авсан хувьсагтай адил ойн санаан хэмжээтэй байна.
* Заалт хүснэгтэнд хийж болохгуй.

# new оператор ашиглан N ширхэг элементтэй бутархай тоон төрлийн хүснэгт үүсгэ. Үүсгэсэн хүснэгтийн элементүүдийн арифметик дунджыг ол.

Example: “2.cpp”

# Ойн цоорхой гэж юу вэ? Жишээгээр тайлбарла.

Санах ойн цоорхой нь програмд хуваарилагдсан санах ойн хэсэг (эсвэл хэсгүүд) үед зөв суллагуй тохиолдолд үүсдэг. Хэдийгээр энэ санах ой нь програмаар ашиглагдахаа больсон ч энэ нь "хэрэглэгдэнэ" гэсэн байгаа учир бусад процесс тэр хэсгийг суллахаас хойш хэрэгэлж чадахгуй учир ихэнхдээ програм гацхад хүрдэг.

void memLeak( )

{

int \*data = new int;

\*data = 15; //\*data заагч устгагдаагуй учир өгөгдлийн мэдэллэл суллагдаагуй, ой хий ажиллаж байна

}

# Delete операторын үүрэг. Давуу тал

New гээр үүссэн объектыг заагч өгсөн бол устгана. Эхлээд destructor дуудаж, дараа нь эзлэж байсан зайг суллана.

Хүснэгт, комлекс объект устгаж байхад оператор нь автоматаар хэмжээ тооцож дотор байгаа бүх юм устана.