

CS200 – Програмчлалын үндэс

Лекц 01

Удиртгал

Профессор А.Эрдэнэбаатар

Хичээлийн тухай

- ▶ Багш: А.Эрдэнэбаатар
 - ▶ Офис: КТМС 104 тоот
 - ▶ Утас: 99119943
 - ▶ E-mail: erka@must.edu.mn
- ▶ Веб сайт:
 - ▶ <http://www.csms.edu.mn/e-learn>

Ашиглах материал

- ▶ Сурах бичиг:
 - ▶ P.J. Deitel and H.M. Deitel, “C How to Program”, Fifth Edition, Pearson Prentice-Hall, 2007.
 - ▶ Brian W. Kernighan and Dennis M. Ritchie, “The C Programming Language”, Second Edition, Prentice Hall, PTR, 1988.
- ▶ Програмчилалын хэрэгсэл:
 - ▶ Dev C++, <http://www.bloodshed.net/devcpp.html>

Хичээлийн агуулга

- ▶ Програмчилалын арга зүй
 - ▶ Алгоритмын зохиомж
 - ▶ Блок схем ба псевдо код
 - ▶ “Дээрээс-доош” програмын зохиомж
- ▶ Бүтцэт хэл (**Си хэл**)
 - ▶ Өгөгдлийн төрөл, Зарлалт, Үйлдэл ба Илэрхийлэл
 - ▶ Оролт ба Гаралт
 - ▶ Удирдлагын бүтцүүд
 - ▶ Функц ба Програмын бүтэц
 - ▶ Заагч
 - ▶ Массив, Мөр ба Нэгдэл
 - ▶ Файлын боловсруулалт
- ▶ Програмчилал
 - ▶ Програмчлах загвар
 - ▶ Бүтцээр кодлох ба Програмын модуль
 - ▶ Програмын баримт бичиг ба Ашиглалт

Лекцийн агуулга

- ▶ Компьютерийн үндсэн ойлголт
- ▶ Програмчилалын хэлний төрлүүд
- ▶ Програмчилалын Си хэл
 - ▶ Түүх
 - ▶ Си стандарт сан
 - ▶ Объект хандалтат програмчилал
 - ▶ Си програм хөгжүүлэх нийтлэг орчин
- ▶ Си хэлний удиртгал

Компьютер гэж юу вэ?

▶ Компьютер

- ▶ Тооцоолол хийх, логик шийдэл гаргах чадвартай төхөөрөмж (техник хангамж)
- ▶ Компьютер нь компьютерийн програм гэгддэг багц удирдамжаар өгөгдөл боловсруулдаг (програм хангамж)

▶ Техник хангамж

- ▶ Төв боловсруулах байгууламж (CPU), эх хавтан, хатуу диск болон бусад (гар, хулгана, дэлгэц, оптик төхөөрөмж г.м.) янз бүрийн төхөөрөмжүүдээс компьютер бүрддэг.
- ▶ Техник хангамжийн хандлага – 1 – 2 жил тутам 2 дахинаар:
 - ▶ Програм дотор нь ажилладаг санах ойн хэмжээ
 - ▶ Програмыг ажиллуулагч процессорын хурд (Мурын хууль)

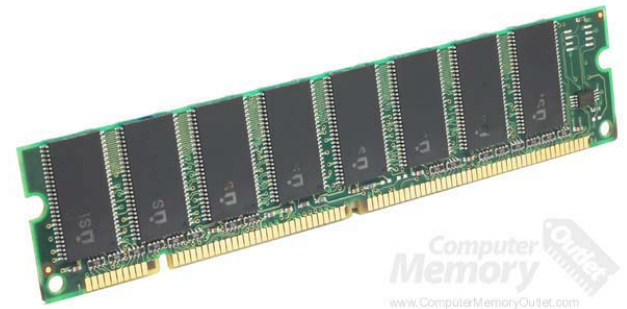
▶ Програм хангамж

- ▶ Компьютер дээр ажиллаж буй програмууд

Компьютерийн зохион байгуулалт

- ▶ Компьютер бүр дараах 6 логик байгууламжтай:
 - ▶ Оруулах
 - ▶ Оролтын төхөөрөмжөөс (гар, хулгана) мэдээлэл авах
 - ▶ Гаргах
 - ▶ Мэдээлэл гаргах (дэлгэц, принтер, бусад төхөөрөмжийг удирдах)
 - ▶ Санах ойн
 - ▶ Шуурхай хандалт, бага багтаамж, оруулсан мэдээллийг хадгалах
 - ▶ Арифметик ба логик (ALU)
 - ▶ Арифметик тооцол болон логик шийдлийг гүйцэтгэдэг
 - ▶ Төв боловсруулах CPU)
 - ▶ Компьютерийн бусад хэсгүүдийг удирдаж, зохицуулдаг
 - ▶ Одоо бол ALU нь түүний үндсэн блок
 - ▶ Хоёрдогч санах ойн
 - ▶ Хямд, удаан хугацааны, их багтаамжтай хадгалагч

Техник хангамж



Computer
Memory
Outlet
www.ComputerMemoryOutlet.com

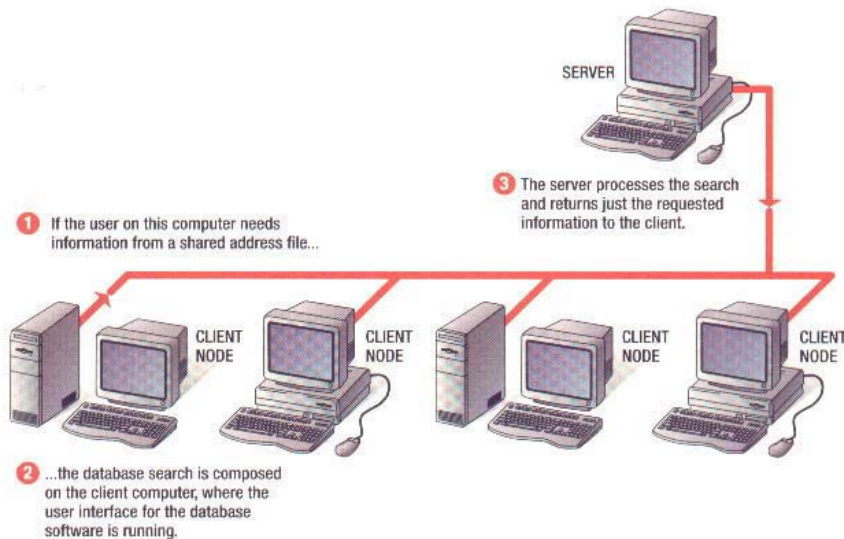
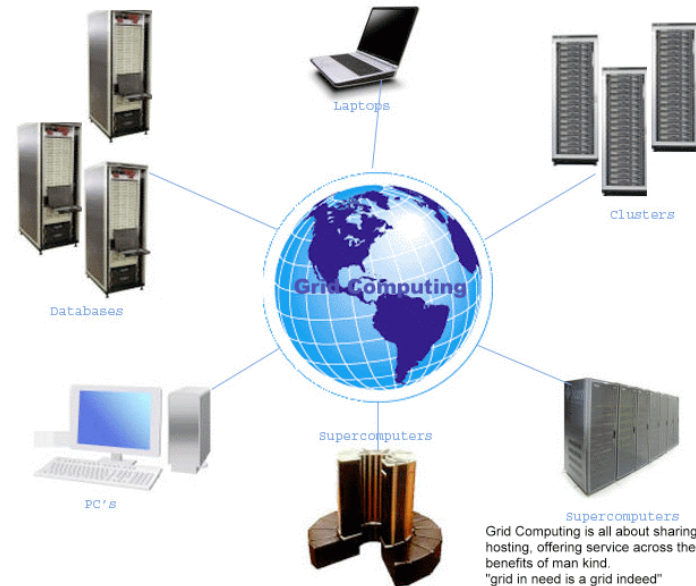
Үйлдлийн системийн түүхээс

- ▶ Багц боловсруулалт
 - ▶ Тухайн агшинд зөвхөн нэг л ажлыг хийдэг
- ▶ Олон програмчилал
 - ▶ Нөөцийн удирдлага, олон програм дундаа ашиглах
 - ▶ Квази програмын зэрэг ажиллагаа
 - ▶ Нэг хэрэглэгч
- ▶ Олон хэрэглэгч/Хугацааны хуваалттай систем
 - ▶ Терминалаар нэгэн зэрэг холбогдсон олон хэрэглэгчийг удирдах
 - ▶ Нөөцийн удирдлага: CPU –ийн төлөвлөлт

Персональ, Тархмал, Клиент/Сервер тооцоолол

- ▶ Персональ тооцоолол
 - ▶ Хувь хүнд эдийн засгийн хувьд тохиромжтой
- ▶ Тархмал тооцоолол
 - ▶ Сүлжээгээр тархсан тооцоолол
- ▶ Клиент/Сервер тооцоолол
 - ▶ Компьютерийн сүлжээн дэх файл сервер болон клиентийн хооронд мэдээллийг хуваан ашиглах

Персональ, Тархмал, Клиент/Сервер тооцоолол



Лекцийн агуулга

- ▶ Компьютерийн үндсэн ойлголт
- ▶ Програмчилалын хэлний төрлүүд
- ▶ Програмчилалын Си хэл
 - ▶ Түүх
 - ▶ Си стандарт сан
 - ▶ Объект хандалтат програмчилал
 - ▶ Си програм хөгжүүлэх нийтлэг орчин
- ▶ Си хэлний удиртгал

Машины, Ассемблер, Дээд түвшний хэл

- ▶ Програмчилалын 3 төрлийн хэл
 - ▶ Машины хэл
 - ▶ Машинаас хамааралтай удирдамжийг CPU гүйцэтгэнэ
 - ▶ Компьютерийн програмын доод түвшний дүрслэл
 - ▶ Ассемблер хэл
 - ▶ Англи маягийн товчлолоор компьютерийн энгийн үйлдлүүдийг дүрсэлдэг (Ассемлер програмаар хөрвүүлэгддэг)
 - ▶ Жишээ: `mov ax, 1234h` (1234h гэсэн утгыг ax регистрт хий)
 - ▶ Дээд түвшний хэл
 - ▶ Ердийн англи хэлтэй төстэй кодтой
 - ▶ Математикийн тэмдэглэгээ ашигладаг (компилятор програмаар хөрвүүлэгддэг)
 - ▶ Жишээ: `grossPay = basePay + overTimePay`

Лекцийн агуулга

- ▶ Компьютерийн үндсэн ойлголт
- ▶ Програмчилалын хэлний төрлүүд
- ▶ Програмчилалын Си хэл
 - ▶ Түүх
 - ▶ Си стандарт сан
 - ▶ Объект хандалтат програмчилал
 - ▶ Си програм хөгжүүлэх нийтлэг орчин
- ▶ Си хэлний удиртгал

Си хэлний түүхээс

▶ Си

- ▶ BCPL болон В хэлнээс Dennis Ritchie гаргаж ирсэн
- ▶ UNIX системийг хөгжүүлэхэд ашигласан
- ▶ Орчин цагийн үйлдлийн системүүдийг бичихэд ашиглагддаг
- ▶ Техник хангамжаас хамааралгүй (авсаархан)

▶ Стандарт

- ▶ Си хэлний хоорондоо зохицдоггүй олон хувилбар байдаг
- ▶ “Амбицгүй, машинаас хамааралгүй” тодорхойлолт гаргах хороо байгуулагдсан
- ▶ 1989 онд стандарт гарсан, 1999 онд сайжруулсан

Си стандарт сан

- ▶ Си програм **функц** гэж нэрлэгддэг модулиас бүрддэг
 - ▶ Програм зохиогч өөрийн функцийг үүсгэж болно.
 - ▶ Давуу тал: програм зохиогч яаж ажиллахыг нь сайн мэддэг
 - ▶ Сул тал: хугацаа шаарддаг
 - ▶ Програм зохиогч “Си сан” –ийн функцийг түлхүү ашигладаг
 - ▶ Тэднийг үндсэн блок болгон ашигладаг
 - ▶ Шинээр дугуй бүтээх хэрэггүй
 - ▶ Хэрэгтэй функц байгаа бол түүнийг өөрөө бичихээс ашиглах нь ерөнхийдөө сайн
 - ▶ Сангийн функц анхааралтай бичигдсэн, бүтээмжтэй, авсаархан байдаг

Лекцийн агуулга

- ▶ Компьютерийн үндсэн ойлголт
- ▶ Програмчилалын хэлний төрлүүд
- ▶ Програмчилалын Си хэл
 - ▶ Түүх
 - ▶ Си стандарт сан
 - ▶ Объект хандалтат програмчилал
 - ▶ Си програм хөгжүүлэх нийтлэг орчин
- ▶ Си хэлний удиртгал

▶ C++

- ▶ Си хэлний дээд түвшин бөгөөд Bell лабораторийн Bjarne Stroustrup зохиосон
- ▶ Си –г “гоёмсог болгож”, объект хандалтат боломжийг суулгасан
- ▶ Объект хандалтат зохиомж бол маш хүчирхэг
 - ▶ Бүтээмж 10 – 100 дахин нэмэгддэг
- ▶ C++ -т суралцах
 - ▶ Нэгэнт C++ нь Си –г өөртөө агуулдаг болохоор Си –г сайн эзэмшээд дараа нь C++ -т суралцах нь дээр

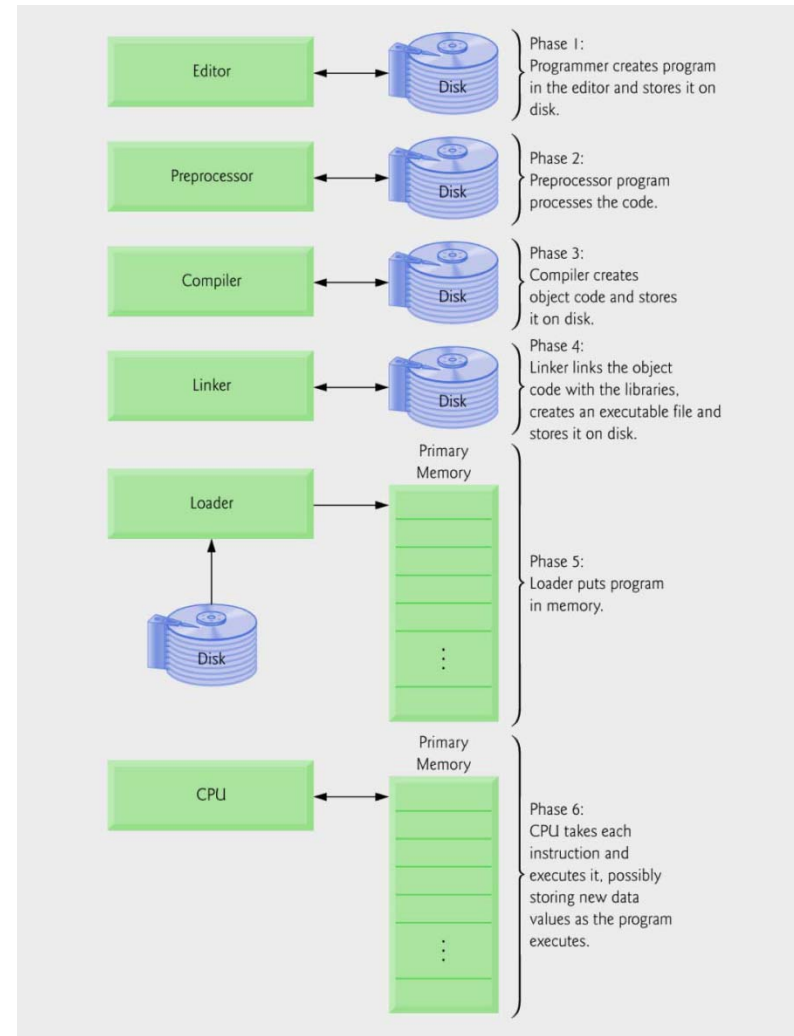
- ▶ Java хэлийг дараах зүйлд голдуу ашигладаг
 - ▶ Динамик, интерактив агуулгатай веб хуудсыг хөгжүүлэх
 - ▶ Том хэмжээний Энтерпрайс програм хөгжүүлэх
 - ▶ Веб серверийн функциональ чанарыг дээшлүүлэх
 - ▶ Хэрэглэгчийн түвшний програм бий болгох (гар утас, пейжер, персонал тоон туслагч г.м.)

Лекцийн агуулга

- ▶ Компьютерийн үндсэн ойлголт
- ▶ Програмчилалын хэлний төрлүүд
- ▶ Програмчилалын Си хэл
 - ▶ Түүх
 - ▶ Си стандарт сан
 - ▶ Объект хандалтат програмчилал
 - ▶ Си програм хөгжүүлэх нийтлэг орчин
- ▶ Си хэлний удиртгал

Си програм хөгжүүлэх нийтлэг орчин

- ▶ Си програмын үе шат
 - ▶ Edit-Засварлах (Си програмын файлын нэр **.c** өргөтгөлтэй)
 - ▶ Preprocess-Препроцесс
 - ▶ Compile-Хөрвүүлэлт
 - ▶ Link-Холболт
 - ▶ Load-Ачаалалт
 - ▶ Execute-Гүйцэтгэл



Лекцийн агуулга

- ▶ Компьютерийн үндсэн ойлголт
- ▶ Програмчилалын хэлний төрлүүд
- ▶ Програмчилалын Си хэл
 - ▶ Түүх
 - ▶ Си стандарт сан
 - ▶ Объект хандалтат програмчилал
 - ▶ Си програм хөгжүүлэх нийтлэг орчин
- ▶ Си хэлний удиртгал

Си хэлний удиртгал

- ▶ Мөр хэвлэх
- ▶ Escape цуваа
- ▶ Хувьсагч, Өгөгдлийн төрөл
- ▶ Санах ойн зарчим
- ▶ Арифметик үйлдэл
- ▶ If-then оператор
- ▶ Дүгнэлт

Энгийн Си програм: Текст мөр хэвлэх

```
01 /* Ex_01 Printing on one line with two printf statements */
02 #include <stdio.h>
03
04 /* function main begins program execution */
05 int main( void )
06 {
07     printf( "Welcome to C!\n" );
08
09     return 0; /* indicate that program ended successfully */
10
11 } /* end function main */
Welcome to C!
```

Тайлбар, түүнийг компьютер алгасна

Препроцессорын удирдамж,
тодорхой файлыг оруулна

Энэ хаалт main функцийн эхлэлийг заана

Энэ оператор хэвлэх
үйлдлийг илэрхийлнэ

Энэ оператораар функц төгсөнө

Энэ хаалт main функцийн төгсгөлийг заана

- Тайлбар
 - /* болон */ -аар хашигдсан текстийг компьютер алгасадаг
 - Програмыг тайлбарлахад ашигладаг
- **#include <stdio.h>**
 - Препроцессорын удирдамж ба тодорхой файлын агуулгыг компьютерт оруулахыг хэлнэ
 - **<stdio.h>** нь стандарт оролт/гаралтын үйлдлийг зөвшөөрч байна

Энгийн Си програм: Текст мөр хэвлэх

▶ **int main()**

- ▶ Си програм нь нэг буюу хэд хэдэн функцтэй байх ба түүний зөвхөн нэг нь үндсэн функц байна
- ▶ Дугуй хаалт '(' ба ')' нь функц гэдгийг илтгэнэ
- ▶ int нь функцээс бүхэл тоон утга буцахыг заана
- ▶ Их хаалт '{' ба '}' нь блокыг илэрхийлнэ. Функцийн их бие ийм хаалтанд бичигдэх ёстой

▶ **printf("Welcome to C!\n");**

- ▶ Компьютерийн гүйэтгэх үйлдэл. Тухайлбал, хашилтанд байгаа тэмдэгт мөрийг хэвлэх
- ▶ Дээрх мөрийг бүхэлд нь оператор гэнэ. Бүх операторууд цэгтэй таслалаар ';' төгсөнө
- ▶ Escape тэмдэг '\'
 - ▶ **printf** ямар нэг тусгай юм гүйцэтгэхийг заана. Тухайлбал, \n нь шинэ мөрийн тэмдэг

Энгийн C програм: Текст мөр хэвлэх

- ▶ **return 0;**
 - ▶ Функцээс гарах арга
 - ▶ Энэ тохиолдолд, програм хэвийн төгссөнийг илтгэнэ
- ▶ Баруун хаалт **}**
 - ▶ **main** функц төгсгөлдөө хүрснийг заана
- ▶ Холбогч
 - ▶ Функцийг дуудахад холбогч түүнийг сангаас олно
 - ▶ Олсон функцээ програмд оруулж өгнө
 - ▶ Хэрвээ функцийн нэр буруу бол холбогч түүнийг сангаас олохгүй, алдаа гарсан тухай мэдээлэл гаргана

Си хэлний удиртгал

- ▶ Мөр хэвлэх
- ▶ Escape цуваа
- ▶ Хувьсагч, Өгөгдлийн төрөл
- ▶ Санах ойн зарчим
- ▶ Арифметик үйлдэл
- ▶ If-then оператор
- ▶ Дүгнэлт

Зарим Escape цуваа

Escape дараалал	Тайлбар
\n	Шинэ мөр. Байршил дараачийн мөрний эхэд очих
\t	Табуляц. Байршил дараачийн табуляцид очих
\a	Авиа. Системийн авиаг гаргах
\\	Эсрэг налуу. Эсрэг налуу тэмдэгтийг оруулах
\"	Давхар хашилт. Давхар хашилт тэмдэгтийг оруулах

Текст мөр хэвлэх програм: өөрөөр

```
01 /* Ex_02 Printing on one line with two printf statements */
02 #include <stdio.h>
03
04 /* function main begins program execution */
05 int main( void )
06 {
07     printf( "Welcome " );
08     printf( "to C!\n" );
09
10     return 0; /* indicate that program ended successfully */
11
12 } /* end function main */
Welcome to C!
```

Энэ printf оператор өмнөх операторын зогссон газраас цааш үргэлжлүүлэн хэвлэнэ

Текст мөр хэвлэх програм: өөрөөр

```
01 /* Ex_03 Printing on one line with two printf statements */
02 #include <stdio.h>
03
04 /* function main begins program execution */
05 int main( void )
06 {
07     printf( "Welcome\nto\nC!\n" );
08
09     return 0; /* indicate that program ended successfully */
10
11 } /* end function main */
```

Шинэ мөрийн тэмдэгт хэвлэлтийг шинэ мөрөнд шилжүүлнэ

```
Welcome
to
C!
```

Си хэлний удиртгал

- ▶ Мөр хэвлэх
- ▶ Escape цуваа
- ▶ Хувьсагч, Өгөгдлийн төрөл
- ▶ Санах ойн зарчим
- ▶ Арифметик үйлдэл
- ▶ If-then оператор
- ▶ Дүгнэлт

Өөр жишээ: 2 тооны нийлбэр

```
01 /* Ex_04 Addition program. Function main begins program execution */
02 #include <stdio.h>
03 int main( void )
04 {
05     int integer1; /* first number to be input by user */
06     int integer2; /* second number to be input by user */
07     int sum;      /* variable in which sum will be stored */
08
09     printf( "Enter first integer\n" ); /* prompt */
10     scanf( "%d", &integer1 );          /* read an integer */
11
12     printf( "Enter second integer\n" ); /* prompt */
13     scanf( "%d", &integer2 );          /* read an integer */
14
15     sum = integer1 + integer2; /* assign total to sum */
16
17     printf( "Sum is %d\n", sum ); /* print sum */
18
19     return 0; /* indicate that program ended successfully */
20
21 } /* end function main */
```

Хувьсагчийн зарлалт

scanf хэрэглэгчээс утга
авч integer1 –т олгоно

scanf хэрэглэгчээс утга
авч integer2 –т олгоно

Нийлбэрийг sum –д
олгоно

```
Enter first integer
45
Enter second integer
72
Sum is 117
```


Өөр жишээ: 2 тооны нийлбэр

► Өмнө үзсэн

► Тайлбар, `#include <stdio.h>`, `main`

► `int integer1, integer2, sum`

- Хувьсагчийн зарлалт, хувьсагч: утга хадгалдаг санах ойн байршил
- `int` гэдэг нь хувьсагч бүхэл тоог хадгалж чадна гэсэн үг (-1, 2, 0, 47)
- Хувьсагчийн нэр (идентификатор). Жишээ нь `integer1`, `integer2`, `sum`. Нэр нь үсэг, цифр(цифрээр эхлэхгүй), доогуур зураас `'_'` –аас бүрдэнэ. (том ,жижиг үсэг ялгаатай)
- Тодорхойлолт гүйцэтгэх оператораас өмнө бичигдэнэ. (гүйцэтгэх оперторт зарлаагүй хувьсагч тааралдвал хөрвүүлэлтийн түвшинд дүрмийн алдааны мэдээлэл гарна)

Өөр жишээ: 2 тооны нийлбэр

- ▶ **`scanf("%d", &integer1);`**
 - ▶ Хэрэглэгчээс утга авна. **`scanf`** нь стандарт оролтыг ашигладаг (голдуу гар)
 - ▶ **`scanf`** хоёр аргументтэй: `"%d"` – өгөгдөл аравтын бүхэл болохыг заана, **`&integer1`** – хувьсагчийн санах ой дахь байршил, **`&`** - одоохондоо **`scanf`** операторт хувьсагчийн өмнө байх ёстой гээд тогтоочих
- ▶ Програм ажиллах явцад **`scanf`** операторын хариу болгож хэрэглэгч тоо бичээд (гарнаас) **`enter/return`** товч дарна

Си хэлний удиртгал

- ▶ Мөр хэвлэх
- ▶ Escape цуваа
- ▶ Хувьсагч, Өгөгдлийн төрөл
- ▶ Санах ойн зарчим
- ▶ Арифметик үйлдэл
- ▶ If-then оператор
- ▶ Дүгнэлт

Санах ойн зарчим

▶ Хувьсагч

- ▶ Хувьсагчийн нэр бүрт компьютерийн санах ойн байршил харгалздаг
- ▶ Хувьсагч бүр нэр, төрөл, хэмжээ, утгатай байна
- ▶ Хувьсагчид шинэ утга (жишээ нь `scanf` -ээр) байршуулах бүрт өмнөх утга солигддог (устгагддаг)
- ▶ Санах ойгоос хувьсагчийг уншихад утга нь өөрчлөгдөхгүй

Санах ойн байршил хувьсагчийн нэр, утгыг харуулж байна

integer1

45

Санах ойн зарчим

Санах ойн байршил хоёр хувьсагчийг оруулны дараа

integer1	45
integer2	72

Санах ойн байршил тооцооллын дараа

integer1	45
integer2	72
sum	117

Си хэлний удиртгал

- ▶ Мөр хэвлэх
- ▶ Escape цуваа
- ▶ Хувьсагч, Өгөгдлийн төрөл
- ▶ Санах ойн зарчим
- ▶ Арифметик үйлдэл
- ▶ If-then оператор
- ▶ Дүгнэлт

Өөр жишээ: 2 тооны нийлбэр

- ▶ = утга олгох үйлдэл
 - ▶ Хувьсагчид утга олгоно
 - ▶ Хоёр гишүүнтэй үйлдэл (+)
 - ▶ `sum = variable1 + variable2;`
 - ▶ Утга авах хувьсагч зүүн талд нь байна
- ▶ `printf("Sum is %d\n", sum) ;`
 - ▶ `scanf` -тэй төсөөтэй, `%d` нь аравтын бүхлийг хэвлэх, `sum` нь ямар бүхлийг хэвлэхийг заана
 - ▶ Тооцооллыг `printf` операторын дотор хийж болно
 - ▶ `printf("Sum is %d\n", integer1 + integer2);`

Арифметик үйлдэл

▶ Арифметик тооцоолол

- ▶ Үржих үйлдэлд '*', хуваах үйлдэлд '/' –г ашиглана
- ▶ Бүхэл тооны хуваалт үлдэгдлийг хаядаг: $7/5$ –аас 1
- ▶ Модуль авах үйлдэл '%' үлдэгдлийг буцаана: $7\%5$ –аас 2 гарна

▶ Үйлдлийн ахлах чанар

- ▶ Зарим үйлдэл бусдаасаа түрүүлж хийгддэг (жишээ үржих нэмэхээс өмнө). Шаардлагатай үед хаалтаар зохицуулна.
- ▶ Жишээ: a , b , c гэсэн 3 хувьсагчийн дундаж
 - ▶ $a + b + c / 3$ – Буруу
 - ▶ $(a + b + c) / 3$ – Зөв

Арифметик үйлдэл

► Арифметик үйлдлүүд

С үйлдэл	Арифметик үйлдэл	Алгебрь илэрхийлэл	С илэрхийлэл
Нэмэх	+	$f + 7$	$f + 7$
Хасах	-	$p - c$	$p - c$
Үржих	*	bm	$b * m$
Хуваах	/	x / y	x / y
Модуль	%	$r \text{ mod } s$	$r \% s$

► Үйлдлийн дарааллын дүрэм

Үйлдэл	Ажиллагаа	Гүйцэтгэх дараалал (ахлах чанар)
()	Хаалт	Түрүүлж хийгдэнэ. Хаалтууд дахарласан бол хамгийн дотор талаас түрүүлж, “нэг түвшингийнх” зүүнээсээ баруун тийш хийгдэнэ
*, /, %	Үржих, Хуваах, Модуль	Хоёрдугаарт хийгдэнэ. Аль алин нь орсон бол зүүнээсээ баруун тийш хийгдэнэ
+, -	Нэмэх, Хасах	Хамгийн сүүлд. Аль алин нь орсон бол зүүнээсээ баруун тийш хийгдэнэ

Жишээ: үйлдлийн дараалал

Алхам 1. $y = 2 * 5 * 5 + 3 * 5 + 7;$
 $2 * 5$ бол 10

Хамгийн зүүн үржих

Алхам 2. $y = 10 * 5 + 3 * 5 + 7;$
 $10 * 5$ бол 50

Хамгийн зүүн үржих

Алхам 3. $y = 50 + 3 * 5 + 7;$
 $3 * 5$ бол 15

Нэмэхээс өмнө үржих

Алхам 4. $y = 50 + 15 + 7;$
 $50 + 15$ бол 65

Хамгийн зүүн нэмэх

Алхам 5. $y = 65 + 7;$
 $65 + 7$ бол 72

Сүүлийн нэмэх

Алхам 6. $y = 72$

Сүүлийн үйлдэл.
 72 –г y –д хийнэ

Си хэлний удиртгал

- ▶ Мөр хэвлэх
- ▶ Escape цуваа
- ▶ Хувьсагч, Өгөгдлийн төрөл
- ▶ Санах ойн зарчим
- ▶ Арифметик үйлдэл
- ▶ If-then оператор
- ▶ Дүгнэлт

Шийдвэр гаргалт: тэнцэтгэл ба харьцааны үйлдэл

- ▶ Гүйцэтгэх оператор
 - ▶ Үйлдэл гүйцэтгэх (тооцоолол, өгөгдөл оруулах/гаргах)
 - ▶ Шийдэл гаргах. Жишээ нь өгсөн тестийн дүнгээс хамаарч “pass” эсхүл “fail” гэсэн үгийг хэвлэх
- ▶ **if** удирдлагын оператор
 - ▶ **if** (хэрэв) нөхцөл үнэн бол **if** операторын их бие гүйцэтгэгдэнэ. Си хэлэнд 0 бол **false** (худал), эсрэг тохиолдолд **true** (үнэн) болдог.
 - ▶ **if** бүтцийн дараа удирдлага үргэлж шилждэг

Шийдвэр гаргалт: тэнцэтгэл ба харьцааны үйлдэл

Алгебрийн тэнцэтгэл, харьцааны үйлдэл	Си хэлний тэнцэтгэл, харьцааны үйлдэл	Си хэлний нөхцөлийн жишээ	Си нөхцөлийн утга
Тэнцэтгэлийн үйлдлүүд			
$=$	<code>==</code>	<code>x == y</code>	x нь y –тэй тэнцүү
\neq	<code>!=</code>	<code>x != y</code>	x нь y –тэй тэнцүү биш
Харьцааны үйлдлүүд			
$>$	<code>></code>	<code>x > y</code>	x нь y –ээс их
$<$	<code><</code>	<code>x < y</code>	x нь y –ээс бага
\geq	<code>>=</code>	<code>x >= y</code>	x нь y –тэй тэнцүү буюу их
\leq	<code><=</code>	<code>x <= y</code>	x нь y –тэй тэнцүү буюу бага

if операторын жишээ

```
01 /* Ex_05 Using if statements, relational
02    operators, and equality operators */
03 #include <stdio.h>
04
05 /* function main begins program execution */
06 int main( void )
07 {
08     int num1; /* first number to be read from user */
09     int num2; /* second number to be read from user */
10
11     printf( "Enter two integers, and I will tell you\n" );
12     printf( "the relationships they satisfy: " );
13
14     scanf( "%d%d", &num1, &num2 ); /* read two integers */
15
16     if ( num1 == num2 ) {
17         printf( "%d is equal to %d\n", num1, num2 );
18     } /* end if */
19
20     if ( num1 != num2 ) {
21         printf( "%d is not equal to %d\n", num1, num2 );
22     } /* end if */
23
24     if ( num1 < num2 ) {
25         printf( "%d is less than %d\n", num1, num2 );
26     } /* end if */
```

num1 нь num2 –той тэнцүү юү?

num1 нь num2 –той тэнцүү биш үү?

num1 нь num2 –оос бага уу?

if операторын жишээ

```
27
28  if ( num1 > num2 ) { ← num1 нь num2 –оос их үү?
29      printf( "%d is greater than %d\n", num1, num2 );
30  } /* end if */
```

```
31
32  if ( num1 <= num2 ) { ← num1 нь num2 –бага буюу тэнцүү юү?
33      printf( "%d is less than or equal to %d\n", num1, num2 );
34  } /* end if */
```

```
35
36  if ( num1 >= num2 ) { ← num1 нь num2 –оос их буюу тэнцүү юү?
37      printf( "%d is greater than or equal to %d\n", num1, num2 );
38  } /* end if */
```

```
39
40  return 0; /* indicate that program ended successfully */
41
42 } /* end function main */
```

Enter two integers, and I will tell you

The relationships they satisfy: 3 7

3 is not equal to 7

3 is less than 7

3 is less then or equal to 7

Enter two integers, and I will tell you

The relationships they satisfy: 22 12

22 is not equal to 12

22 is greater than 12

22 is greater then or equal to 12

Өмнө үзсэн үйлдлүүдийн дараалал

Үйлдэл	Шийдэх чиглэл
()	Зүүнээс баруун
* / %	Зүүнээс баруун
+ -	Зүүнээс баруун
< <= > <=	Зүүнээс баруун
== !=	Зүүнээс баруун
=	Баруунаас зүүн

Си хэлний дүрмээр нөөцлөгдсөн
түлхүүр үгсийг идентификатор
буюу хувьсагчийн нэрэнд
ашиглаж **БОЛОХГҮЙ**

Си хэлний түлхүүр үгс							
auto	const	double	float	int	short	struct	unsigned
break	continue	else	for	long	signed	switch	void
case	default	enum	goto	register	sizeof	typedef	volatile
char	do	extern	if	return	static	union	while

Си хэлний удиртгал

- ▶ Мөр хэвлэх
- ▶ Escape цуваа
- ▶ Хувьсагч, Өгөгдлийн төрөл
- ▶ Санах ойн зарчим
- ▶ Арифметик үйлдэл
- ▶ If-then оператор
- ▶ Дүгнэлт

Дүгнэлт

- ▶ Тайлбар нь `/*` -аар эхэлж `*/` -аар төгсөнө. Тайлбар програмыг баримтжуулж, уншигдах байдлыг сайжруулдаг.
- ▶ `#include` удирдамж препроцессорт өөр файлын агуулгыг оруулахыг хэлж өгдөг. (ихэвчлэн `<stdio.h>` мэтийн толгой файл)
- ▶ `<stdio.h>` толгой файл стандарт оролт/гаралтын сангаас `printf` мэтийн функцийг дуудсан тухай мэдээллийг хөрвүүлэгчид хэлж өгдөг.
- ▶ Ямар ч Си програм `main` функцээс эхэлж хэрэгждэг.
- ▶ Функц нь мэдээлэл буцааж (`return`) болдог.
- ▶ Функц мэдээлэл авч болно.
- ▶ Функц `{` хаалтаар эхэлж `}` хаалтаар төгсдөг.
- ▶ Оператор бүрийн төгсгөлд цэгтэй таслал `;` тавина.
- ▶ Шинэ мөр `\n` мэтийн escape тэмдэгтийг хэрэглэж болно.

Дүгнэлт

- ▶ Хувьсагч гэдэг бол програм утга хадгалах зорилгоор ашигладаг санах ойн байршил юм.
- ▶ Хувьсагч бүрийг програм ашиглаж эхлэхээс нь өмнө нэр, төрлийг нь зааж **зарлах** ёстой.
- ▶ **%** нь өгөгдөл аравтын бүхэл тоо гэдгийг илтгэдэг.
- ▶ **&** нь хувьсагчийн санах ойн хаяг гэдгийг илтгэдэг.
- ▶ **printf** функц мөрийн хэвшүүлэлтийг удирдах мөрийг ашиглаж болдог.
- ▶ Хувьсагч бүр нэр, төрөл, утгатай байна. Хувьсагчийн нэр нь компьютерийн санах ой дахь харгалзах байршлыг заадаг.
- ▶ Арифметик болон бусад үйлдлүүд нь үйлдлийн дараалалыг дагадаг.

Дүгнэлт

- ▶ **if** оператор нь **нөхцөл** гэж нэрлэгддэг фактын үнэн, худалд суурилж шийдвэр гаргах боломжийг програмд олгодог.
- ▶ **if** операторт нөхцлийг үүсгэхдээ тэнцэтгэл болон харьцааны үйлдлийг ашигладаг.
- ▶ Харьцааны үйлдлүүд ч үйлдлийн ахлах чанартай байна.
- ▶ **=** -ийг авах, **==** -ийг “тэнцүү” гэж уншдаг.
- ▶ Си хэлэнд тодорхой тооны түлхүүр үгийг ашигладаг ба тэдгээрийг өөр зорилгоор ашиглаж болохгүй.