

1 Хүснэгт

Санах ойд дараалан байрлах элементүүдийн (хувьсагчуудын) цувааг хүснэгт гэж нэрлэдэг.

1.1 Нэг хэмжээст хүснэгт

Дүрэм:

төрөл нэр[хэмжээ];

Хүснэгт нь бидний төсөөлж байгаа мөр, баганатай хүснэгт мөн. Дээрх хүснэгт нь зөвхөн 1 мөртэй, "хэмжээ" ширхэг баганатай байна. Үүнийг **нэг хэмжээст хүснэгт** гэдэг.

Жишээ:

```
1 int a[100]; // 100 хэмжээтэй хүснэгт
2 double d[20]; // 20 хэмжээтэй хүснэгт
```

Олон хэмжээст хүснэгт

Хэд хэдэн мөрөөс тогтох хүснэгтийг хоёр хэмжээст хүснэгт гэдэг. Хоёр хэмжээст хүснэгт мөр болон баганаас бүрдэнэ.

Дүрэм:

төрөл нэр[мөрийн_тоо][баганы_тоо];

Жишээ:

```
1 int a[100][100];
```

Дээрх хүснэгт нийтдээ 100×100 ширхэг тоо хадгалах багтаамжтай ба санах ойд $\text{sizeof}(\text{int}) \times 100 \times 100$ байт зайг эзэлнэ.

Хоёр хэмжээст хүснэгтийн эхний мөрийн элементүүд санах ойд дараалан байрлаад дараагаар нь дараагийн мөрийн элементүүд санах ойд залгуулан байрлана.

Ерөнхий тохиолдолд та хэдэн ч хэмжээст хүснэгтийг байгуулж болно.

```
1 int a[10][10][10][10]; // 4 хэмжээст хүснэгт
```

Хүснэгтийг файлын хүрээнд зарлагдвал хэмжээ нь заавал тогтмол байна.

```
1 #include <stdio.h>
2 int size = 10;
3 int array[size]; // Aldaa: filiin hureend huvisah хэмжээтэй хүснэгт
4 char s[100]; // Zov: togtmol хэмжээтэй хүснэгт
5 int main() {
6     int n;
7     scanf("%d", &n);
8     int a[n]; // Zov: n хэмжээтэй хүснэгт main() функts дотор.
9     return 0;
10 }
```

Хүснэгтийг эхлүүлэх

Хүснэгтийг зарлах үед шууд тоонуудаар эхлүүлж болдог.

- Бүх элементийг тоогоор эхлүүлэх

```
1 int a[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

- Орхигдсон дугааруудын утга 0-ээр эхлүүлэгдэнэ.

```
1 int a[10] = {1, 2}; // 1, 2, 0, 0, 0 ... gej heluulegdene
```

- Бүх элементүүдийг 0-ээр эхлүүлэх

```
1 int a[10] = {0}; // bugd 0
```

- Хүснэгтийн хэмжээ байхгүй бол эхлүүлсэн утгуудын тоогоор тодорхойлогдоно.

```
1 int a[] = {0, 1, 2}; // 3 hemjeetei husnegt
```

- Хоёр хэмжээст хүснэгтийг эхлүүлэх

```
1 int move[][2] = {
2     {1, 0},
3     {0, 1},
4     {-1, 0},
5     {0, -1}, // Gurvan morteи husnegt
6 };
```

1.2 Хүснэгтийн элементүүдэд хандах

N хэмжээтэй хүснэгтийн индекс болох i тоо нь $0 \leq i < N$ байна. Өөрөөр хэлбэл индекс тоонуудын ялгаатай утгууд нь хүснэгтийн цор ганц элементийг заана. Хүснэгтийн индексийг хэтрүүлэн хандахад ажиллах үеийн алдаа үүснэ.

Жишээ:

```
1 int a[100];
2 int i, n = 10;
3 for (i = 0; i < n; i++)
4     scanf("%d", &a[i]); // Garaas a[i]-d utga unshij bna.
5
6 a[3] = 0; // Xusnegtiin 4 daxи elemented 0 utga onoono
7 // husnegtee xevley
8 for (i = 0; i < n; i++)
9     printf("%d ", a[i]); // Garaas a[i]-d utga unshij bna.
10 printf("\n");
11 a[100] = 3; // Aldaa: index 99-oos hetreh ysgui
```

Хүснэгтийн элементүүд санах ойд дараалан байрлана гэдэг нь элементүүд нь хэрэв хүснэгт *int* төрөл бол дөрөв, дөрвөн байтын зайтай, харин *char* төрөл бол нэг, нэг байтын зайтай санах ойд байрлана гэсэн үг.

Доорх жишээнээс хүснэгтийн элементүүдийн хаягийн зөрөөг ажиглана уу.

```

1 #include <stdio.h>
2 int main() {
3     int ai[100];    // int husnegt zarlaj bna
4     char ac[100];  // char husnegt zarlaj bna
5     printf("%llu\n", &ai[0]); // 0-р элементийн хаяг
6     printf("%llu\n", &ai[1]); // 1-р элементийн хаяг
7     printf("%llu\n", &ai[2]); // 2-р элементийн хаяг
8     printf("%llu\n", &ac[0]); // 0-р элементийн хаяг
9     printf("%llu\n", &ac[1]); // 1-р элементийн хаяг
10    printf("%llu\n", &ac[2]); // 2-р элементийн хаяг
11    return 0;
12 }

```

Санах ойн хаяг нь 64 битийн програм гаргасан бол тэмдэггүй 8 байт тоо байдаг тул unsigned long long төрөл байх ёстой. Уг төрлийг хэвлэхдээ "%llu" форматыг ашиглана.

1.3 Дасгалууд

Ангид

- 1-с 10 хүртэлх тоог хүснэгтэд хадгалж. Тэдгээрийг хэвлэн харуул.
- $1 \leq n < 100$ тоо, дараа нь дагалдах n ширхэг тоонуудыг хэрэглэгчээс уншин хүснэгтэд хадгалж, хүснэгтээсээ оруулсан дарааллын эсрэгээр хэвлэн харуул.

Оролт:

5

3 7 1 8 2

Гаралт:

2 8 1 7 3

- Эхний мөрөнд $1 \leq n, m < 100$ хоёр тоо өгөгдөнө. Дараагийн мөрөнд өгөгдөх n ширхэг тоонуудыг хэрэглэгчээс уншин A хүснэгтэд хадгална. Гурав дахь мөрөнд m ширхэг тоонуудыг хэрэглэгчээс уншин B хүснэгтэд хадгална. Тэгвэл A хүснэгтэд байгаа n ширхэг тоо, B хүснэгтэд байгаа m ширхэг тоог нийлүүлэн C хүснэгтэд хадгал.
- $1 \leq n < 100$ тоо, дараа нь дагалдах n ширхэг тоонуудыг хэрэглэгчээс уншин хүснэгтэд хадгалж, дараагаар нь x тоог хэрэглэгчээс уншин тухайн хүснэгтэд байгаа эсэхийг шалган байрлалын дугаарыг хэвлэн харуул, байхгүй бол -1 -г хэвлэ. Олон удаа олдож байвал байрлал бүрийг хэвлэнэ.
- $1 \leq n < 100$ тоо, дараа нь дагалдах n ширхэг тоонуудыг хэрэглэгчээс уншин хүснэгтэд хадгалж өсөхөөр эрэмбэлэгдсэн, буурахаар эрэмбэлэгдсэн, эсвэл эрэмбэлэгдээгүй болохыг тогтоо.
- $1 \leq n, m < 100$ тоонууд, дараа нь дагалдах $n \cdot m$ ширхэг тоонуудыг хэрэглэгчээс уншин A хоёр хэмжээст хүснэгтэд хадгалж, дараагаар нь x тоог хэрэглэгчээс уншин тухайн хүснэгтийн хэд дүгээр мөр, хэд дүгээр баганад байгааг хэвлэн харуул. Байхгүй бол 1, -1"гэж гаргана уу. Олон удаа олдож байвал байрлал бүрийг хэвлэнэ.

Нэмэлт бодлого

1. $1 \leq n < 100$ тоо, дараа нь дагалдах n ширхэг тоонуудыг хэрэглэгчээс уншин A хүснэгтэд хадгалж, $1 \leq m < 100$ тоо, дараа нь дагалдах m ширхэг тоонуудыг хэрэглэгчээс уншин B хүснэгтэд хадгалж, A хүснэгтийн ямар ямар элемент B хүснэгтэд байгаа болохыг хэвлэн харуул.

Гэрт

1. Тус бүр нь N , M хэмжээтэй A , B хоёр хүснэгтийг гараас авч, C хүснэгтэд $A \cup B$ олонлогийн элементүүдийг хадгалан хэвлэ. Өөрөөр хэлбэл A -д байх бүх элементийг бас B -д байх бүх элементийг давхардуулалгүйгээр C хүснэгтэд хадгалж хэвлэн харуул. C хүснэгтэд хадгалагдах тоонууд ямар ч дарааллаар хэвлэгдэж болно.

Оролт:

5 3

3 7 1 3 2

1 5 3

Гаралт:

7 2 5 1 3

2. Гараас N тоо, мөн $N \times N$ хэмжээтэй хүснэгт (хоёр хэмжээст хүснэгт) уншин мөр бүрийн нийлбэр, багана бүрийн нийлбэр, зүүн, баруун диагоналиудынх нь нийлбэрийг тус тус хэвлэ.
3. Давхар бүрдээ M ширхэг айлтай, K орцтой N давхар байрны айл болгоныг дугаарла. Айлын дугаарыг 1-с эхлэн орц орцоор нь 3 хэмжээст хүснэгтэд хадгалан хэвлэн харуул.