

### 1-masala.

Tasavvur qiling-a, sizda berilgan naqshga muvofiq turli xil ranglar bilan to'ldirishingiz kerak bo'lgan kvadratchalar qatori bor. Kvadratchalarni ketma-ket bo'yash kerak, ya'ni keyingi kvadrat boshqa rangda bo'lsa, qalamni o'zgartirishingiz kerak bo'ladi.

**Eslatma: Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.**

Ranglar ro'yxatini oladigan va butun naqshni to'ldirish uchun zarur bo'lgan vaqtning qiymatini (soniyalarda) qaytaradigan funksiyani yozing.

Bunda:

- qalamni almashtirish uchun 1 soniya kerak bo'ladi
- kvadratni to'ldirish uchun 2 soniya kerak bo'ladi

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
naqsh(["Red", "Blue", "Red", "Blue", "Red"])	14
naqsh(["Blue"])	2
naqsh(["Red", "Yellow", "Green", "Blue"])	11
naqsh(["Blue", "Blue", "Blue", "Red", "Red", "Red"])	13

### 2-masala.

6090609 sonining o'ziga xos xususiyati bor: agar siz uni teskari aylantirsangiz (ya'ni 180 gradusga aylantirsangiz), xuddi shu son kelib chiqadi - 6090609

**Eslatma: Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.**

0, 6, 9 raqamlar ishtirok etgan qatorni kirituvchi dasturni yozing va hosil qilingan son teskari aylantirilgandan keyin bir xil son bo'lish yoki bo'lmasligini aniqlang.

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
6090609	True
9669	False
69069069	True
69	True

### 3-masala.

Foydalanuvchi satr kiritadi. Kiritilgan satr namunaviy telefon raqamiga mos ekanligini tekshiring.

Namunaviy haqiqiy raqam: (998) 97-456-78-90.

**Eslatma: Yopuvchi qavsdan keyin bo'sh joy qo'yishni unutmang. Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.**

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
(998) 98-123-45-67	True
(998) 90-12-12-12	False
998) 95-987-65-43	False
(998) 93-654-12-78	True

#### 4-masala.

Berilgan ism (name) va yosh (age) asosida to'rtburchaklardan "tort" yaratadigan dastur yozing. Tort faylda 3 qatordan tashkil topadi. "Tug'ilgan kuningiz bilan" jumlasini yozilgan tort belgisi bilan o'ralgan:

- agar yosh juft son bo'lsa, ramka "#" belgilaridan iborat
- agar yosh toq son bo'lsa, ramka "\*" belgilaridan iborat.

#### Ko'rsatmalar:

Tabriklash quyidagi formatda bo'lishi kerak:

{age} Happy Birthday {name}! {yosh}

Ramka va yosh raqamlari o'rtasida bo'sh joy bo'lishi kerak.

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
name="Jack" age=10	"#####" "# 10 Happy Birthday Jack! 10 #" "#####"
name="Russell" age=19	"*****" "* 19 Happy Birthday Russell! 19 *" "*****"
name="Isabelle" age=2	"#####" "# 2 Happy Birthday Isabelle! 2 #" "#####"

#### 5-masala.

User nomli jadvalda er va xotinlar, shuningdek, uylanmagan / turmush qurmaganlar bor. Hamma uchun ism, familiya, yosh, manzil ko'rsatilgan. Er va xotin bir xil familiyaga ega va bir xil manzilda yashaydilar. Ushbu jadval **id**, **name**, **surname**, **age**, **adress** ustunlaridan iborat.

Masalan,

id	name	surname	age	address
1	Abdulla	Abdullayev	25	Toshkent sh., M.Ulug‘bek tumani, Bobur ko‘chasi 5-uy
2	Nodira	Abdullayeva	22	Toshkent sh., M.Ulug‘bek tumani, Bobur ko‘chasi 5-uy
3	Shoxrux	Rustamov	20	Toshkent sh., Chilonzor tumani, Namatak ko‘chasi 1-uy 25-honadon
4	Hilola	Rasulova	21	Toshkent sh., Chilonzor tumani, Jalilov ko‘chasi 5-uy 2-honadon
5	Jalil	Rasulov	18	Toshkent sh., Chilonzor tumani, Jalilov ko‘chasi 5-uy 2-honadon
6	Temur	Qodirov	23	Toshkent sh., Yashnobod tumani, Safiya ko‘chasi, 25-uy
7	Sardor	Kamolov	20	Toshkent sh., Yunusobod tumani, 2-kvartal, 4-uy 75-honadon
8	Botir	Malikov	19	Toshkent sh., Mirobod tumani, Temuriylar ko‘chasi, 12-uy
9	Lola	Kamolova	24	Toshkent sh., Yunusobod tumani, 2-kvartal, 4-uy 75-honadon
10	Maftuna	Farruxova	19	Toshkent sh., Olmazor tumani, Amir Temur ko‘chasi, 4-uy 63-honadon

**User** jadvali ismi (**name**), familiyasi (**surname**), yoshi (**age**) va manzili (**adress**) lardan tashkil topgan va ularning ichidagi ma’lumotlarni to’ldiring.

1-vazifa: **User** jadvalidan foydalanib, oilali foydalanuvchilar haqida ma’lumotlarni chiqaring.

2-vazifa: **User** jadvalidan foydalanib, oila qurmagan foydalanuvchilar haqida ma’lumotlarni chiqaring.

## 6-masala.

Fermer quyida berilgan **st** qatorida yashiringan hayvonlar sonini bilish uchun sizning yordamingizga muhtoj.

Bizda hayvonlarning nomlaridan iborat ro'yxat mavjud:

```
animals = ["dog", "cat", "bat", "cock", "cow", "pig",
           "fox", "ant", "bird", "lion", "wolf", "deer", "bear",
           "frog", "hen", "mole", "duck", "goat"]
```

Funksiya parametri sifatida **st** satrini oladigan va satr belgilaridan foydalanib hosil qilish mumkin bo'lgan hayvon nomlarining maksimal sonini va hayvonlarning nomini chiqaradigan funksiya yozing.

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
st = "goatcode"	count_animals(st) → 2 (dog, goat)
st = "cockdogwdufrbir"	count_animals(st) → 4 (cow, duck, frog, bird)
st = "dogdogdogdogdog"	count_animals(st) → 5 (dog)

### 7-masala.

Labirintni ikki o'lchovli matritsa sifatida ko'rsatish mumkin, bunda **nol(0)**lar yurish mumkin bo'lgan maydonlarni, **bir(1)**lari esa devorlarni ifodalaydi. Siz yuqori chap burchakdan o'yinni boshlaysiz va chiqish pastki o'ng burchakda joylashgan.

Agar labirintning bir chetidan kirib, ikkinchisidan chiqish mumkin bo'lsa, **True**, aks holda **False** chiqarish kerak. Siz faqat yuqoriga, pastga, chapga va o'ngga harakat qilishingiz mumkin. Siz diagonal bo'ylab harakatlana olmaysiz.

Ikki o'lchovli matritsaning uzunligini 5 ta qator va 7 ta ustun deb olinsin va foydalanuvchi tomonidan kiritilsin.

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
[ [0, 1, 1, 1, 1, 1, 1], [0, 0, 1, 1, 0, 1, 1], [1, 0, 0, 0, 0, 1, 1], [1, 1, 1, 1, 0, 0, 1], [1, 1, 1, 1, 1, 0, 0] ]	True
[ [0, 1, 1, 1, 1, 1, 1], [0, 0, 1, 0, 0, 1, 1], [1, 0, 0, 0, 0, 1, 1], [1, 1, 0, 1, 0, 0, 1], [1, 1, 0, 0, 1, 1, 1] ]	False
[ [0, 1, 1, 1, 1, 0, 0], [0, 0, 0, 0, 1, 0, 0], [1, 1, 1, 0, 0, 0, 0], [1, 1, 1, 1, 1, 1, 0], [1, 1, 1, 1, 1, 1, 1] ]	False
[ [0, 1, 1, 1, 1, 0, 0], [0, 0, 0, 0, 1, 0, 0], [1, 1, 1, 0, 0, 0, 0], [1, 0, 0, 0, 1, 1, 0], [1, 1, 1, 1, 1, 1, 0] ]	True

## 1-masala. DOMINO

Tasavur qiling, siz stolda **domino** o‘yinida paydo bo‘lib qoldingiz. Stolda **dominon**ing bir nechta donalari yotibdi. Sizning vazifangiz ushbu stoldagi yetishmayotgan **domino** donalarini chiqarish.

**Domino** donalari:




[0,0], [0,1], [0,2], [0,3], [0,4], [0,5], [0,6], [1,1], [1,2], [1,3], [1,4], [1,5], [1,6], [2,2], [2,3], [2,4], [2,5], [2,6], [3,3], [3,4], [3,5], [3,6], [4,4], [4,5], [4,6], [5,5], [5,6], [6,6]

**Eslatma: Stolda domino donalari teskari yotishi ham mumkin. Barcha ma’lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo‘yiladi.**

Input (Kiruvchi ma’lumot)	Output (Chiquvchi ma’lumot)
[[1,6], [6,3], [0,5], [5,2], [4,1], [3,5], [5,1], [4,5], [5,5], [2,6], [6,0], [0,1], [0,0], [2,2], [1,1], [1,3], [4,6], [4,4]]	[0,2] [0,3] [0,4] [1,2] [2,3] [2,4] [3,3] [3,4] [5,6] [6,6]

## 2-masala. DONUTS

“Ba’zida nollar juda mazali **donut**larga o‘xshaydi va har safar **donut**ni tugatganimizda, biz ya’na boshqasini, keyin boshqasini va ya’na boshqasini xohlaymiz ...”

Sizga butun sonlardan iborat **List** beriladi. Ushbu missiyadagi vazifangiz berilgan **List**dagi barcha nollarni (... ,  , ... --> ...,  ,  , ...) ko‘paytirish (donut haqida o‘ylab ko‘ring) kerak. Keling, bir misolni ko‘rib chiqaylik:

[1, 0, 2, 0] -> [1, 0, 0, 2, 0, 0]

Sizning vazifangiz **duplicate\_zeros** nomli funksiya yarating va ushbu funksiya **List** ma’lumotlarini qabul qilsin va ushbu **List**dagi ma’lumotlar orasidagi nollarni ikkilantirsin(ya’ni dublikatini yaratsin).

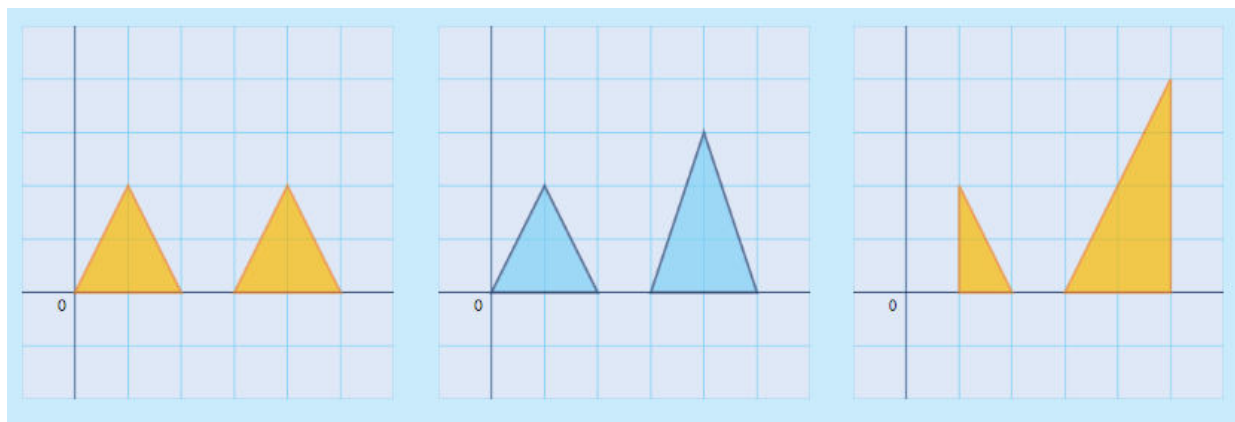
**Eslatma: Barcha ma’lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo‘yiladi.**

Input (Kiruvchi ma’lumot)	Output (Chiquvchi ma’lumot)
duplicate_zeros([1, 0, 2, 3, 0, 4, 5, 0]))	[1, 0, 0, 2, 3, 0, 0, 4, 5, 0, 0 ]
duplicate_zeros([0, 0, 0, 0]))	[0, 0, 0, 0, 0, 0, 0, 0]
duplicate_zeros([100, 10, 0, 101, 1000]))	[100, 10, 0, 0, 101, 1000]

### 3-masala. EQUAL TRIANGLES

Bu ikki uchburchakning o'xshashligini tekshirish vazifasi.

Sizga har bir uchburchakning uchlari koordinatalari sifatida ikkita List ma'lumotlari beriladi. Siz chiquvchi ma'lumot sifatida boolni qaytarishingiz kerak. (Uchburchaklar o'xshash yoki yo'q)



**Eslatma:** Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
Trian1 = [(0, 0), (1, 2), (2, 0)] Trian2 = [(3, 0), (4, 2), (5, 0)]	True
Trian1 = [(0, 0), (1, 2), (2, 0)] Trian2 = [(3, 0), (4, 3), (5, 0)]	True
Trian1 = [(1, 0), (1, 2), (2, 0)] Trian2 = [(3, 0), (5, 4), (5, 0)]	True

### 4-masala. C/C++ COMPILER

Faylda C dasturlash tilida yozilgan kod bor. Ushbu kodni fayldan o'qib olib unda **o'zgaruvchi=o'zgaruvchi+1** qismini **o'zgaruvchi++** ga almashtiring. (O'zgaruvchining nomi faqat bitta harfdan iborat deb hisoblansin)

**Eslatma:** Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.

Input (Fayldagi ma'lumot)	Output (Faylga yangi yozilgan ma'lumot)
<pre>#include &lt;stdio.h&gt; int main() {     int a=5;     a=a+1;     int b=a*a;     b=b+1;     printf("a=%d\tb=%d\n",a,b);     return 0; }</pre>	<pre>#include &lt;stdio.h&gt; int main() {     int a=5;     a++;     int b=a*a;     b++;     printf("a=%d\tb=%d\n",a,b);     return 0; }</pre>

## 5-masala. CITY\_INFORMATION

MySQL serverda **city** nomli jadval tuzing. Bu jadval o'zida quyida keltirilgan ustunlarni saqlashi lozim:

Ustun nomi	Tipi	Tavsif
id	INTEGER AUTO_INCREMENT	ID raqam
name	VARCHAR(17)	Shahar nomi
countrycode	VARCHAR(3)	Mamlakatning qisqacha kodi (masalan, UZB yoki RUS)
population	INTEGER	Aholi soni

Maqsadingiz shu bazani kamida 10 ta ma'lumotlar bilan to'ldirib, quyidagi shartlarga javob beruvchi so'rovlarni yozish:

- aholisi 1milliongacha bo'lgan va 1 milliondan ko'p bo'lgan shaharlar sonini chiqaring.(alohida ustunlarda chiqishi kerak)
- O'zbekiston davlatiga tegishli shaharlarni va ularni aholi sonini chiqaring.

**Eslatma: Barcha ma'lumotlarni CREATE va INSERT INTO orqali kodda yozishingiz shart!**

## 6-masala. BOOK

**Book** nomli class yarating va uning property elementlari quyidagilardan iborat:

- Name;
- Count\_pages;
- Price.

Ushbu classga tegishli 5ta obyekt yarating va ularni ma'lumotlarni kiriting.

Sizning vazifangiz quyidagilardan iborat:

- 1) Barcha kitoblar orasida bitta sahifaning o'rtacha narxini aniqlang.
- 2) Kitoblar nomida "Programmimg" so'zi ishtirok etgan barcha kitoblarni narxini 2 barobarga oshiring.

**Eslatma: Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.**

## 7-masala. PHARMACY

**Date** nomli class e'lon qiling va uning propertylari 3ta butun son(**day, month, year**)dan iborat. **Drug** nomli class **Date** classidan voris bo'lib keladi va uning propertylari **drug\_name, date\_create** va **company\_name** lardan iborat.

**Drug** classiga tegishli 1ta obyekt yarating va ularni ma'lumotlarini kiriting.

Sizning vazifangiz ushbu dorining yaratilgan sanasidan boshlab hozirgi paytgacha nechki kun bo'lganligini aniqlaydigan methodni yarating.

**Eslatma: Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.**

## 8-masala. TRAIN

Afrosiyob poyezdi yo'l bo'ylab **N**ta bekatdan o'tishi kerak (1-boshlang'ich bekat, **N**-oxirgi bekat hisoblanadi). Ushbu poyezdagi yo'lovchilar ro'yhati berilgan. Bundan tashqari ularni qaysi bekatda kirishi va qaysi bekatda chiqishi ma'lum. Qaysi qatnovlarda(ya'ni qaysi qo'shni stansiyalar orasida) eng ko'p yo'lovchi bo'lgan.

### Kiruvchi ma'lumot:

Birinchi qatorda **N** – bekatlar soni kiritiladi. Ikkinchi qatorda **M** – yo'lovchilar soni va undan keyingi qatorda esa **M**ta yo'lovchi haqida ma'lumot quyidagi tartibda kiritiladi:

**Ismi kirish\_bekati\_soni chiqish\_bekati\_soni (1dan N+1gacha sonlar)**

### Chiquvchi ma'lumot:

Sizning vazifangiz poezdda eng ko'p yo'lovchilar bo'lgan qatnovlar ro'yxatini chiqarish kerak. Ushbu ma'lumotda har bir qo'shni bekatlar '-' belgisi bilan ajratilgan bo'lishi kerak.

**Eslatma: Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.**

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
N=5	1-2
M=3	3-4
Sanjar 1 5	4-5
Baxrom 3 5	
Kamola 1 2	



## 9-masala. ELECTIONS

Guruh sardorini saylash marosimida ovozlarning yarmidan ko‘pini to‘plagan nomzod g‘alaba qozonadi. Agar bunday nomzod bo‘lmasa, eng ko‘p ovoz to‘plagan ikki nomzod saylovning ikkinchi bosqichiga o‘tadi.

### Kiruvchi ma’lumot:

Kirish faylining har bir satrida bitta saylovchi ovoz bergan nomzodning ismi ko‘rsatilgan. Ma’lumki, nomzodlarning umumiy soni 20 nafardan oshmaydi, biroq nomzodlar ro‘yxati aniq ko‘rsatilmagan.

### Chiquvchi ma’lumot:

Agar 50% dan ortiq ovoz olgan nomzod bo‘lsa, dasturda uning ismi ko‘rsatilishi kerak. Agar bunday nomzod bo‘lmasa, dasturda birinchi o‘rinni olgan nomzodning ismi, keyin ikkinchi o‘rinni olgan nomzodning ismi ko‘rsatilishi kerak.

**Eslatma: Barcha ma’lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo‘yiladi.**

Input (Faylning ichidagi ma’lumot)	Output (Chiquvchi ma’lumot)
Sanjar Botir Sanjar	Sanjar
Sanjar Farrux Go‘zal Farrux Farrux Sanjar	Farrux Sanjar

### 10-masala. QUEENS

Ma'lumki, 8 ta Farzin(Qirolicha) bir-biriga hujum qilmasligi uchun  $8 \times 8$  o'lchamdagi taxtaga joylashtirilishi mumkin. Duskada sizga 8 ta Farzin(Qirolicha)dan iborat tartib berilgan, ularning bir-birini uradigan juftligi bor yoki yo'qligini aniqlang.

#### Kiruvchi ma'lumot:

Dastur kirish sifatida sakkiz juft raqamlarni oladi, 1 dan 8 gacha bo'lgan har bir raqam 8 ta Farzin(Qirolicha)ning koordinatalari.

#### Chiquvchi ma'lumot:

Agar qirolichalar bir-birini urishmasa, **NO** so'zini chop eting, aks holda **YES** so'zini chop eting.

**Eslatma:** Barcha ma'lumotlarni foydalanuvchi kiritishi lozim va algoritmsiz ishlangan misolga past ball qo'yiladi.

Input (Kiruvchi ma'lumot)	Output (Chiquvchi ma'lumot)
1 7 2 4 3 2 4 8 5 6 6 1 7 3 8 5	NO
1 8 2 7 3 6 4 5 5 4 6 3 7 2 8 1	YES