

প্রবলেম - 1

কোনো একটি পূর্ণসংখ্যা দেওয়া থাকলে সেটি জোড় না বিজোড় তা বের করতে হবে।

ইনপুট

প্রথম লাইনে একটি সংখ্যা T ($1 \leq T \leq 100$) দেওয়া থাকবে। পরবর্তীতে T এর মান যত, ততটি লাইন থাকবে। প্রতিটি লাইনে একটি করে পূর্ণসংখ্যা n ($0 \leq n \leq 2147483647$) দেওয়া থাকবে।

আউটপুট

প্রতিটি পূর্ণসংখ্যার জন্য, সংখ্যাটি জোড় হলে even আর বিজোড় হলে odd কথাটি প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
100
0
1111
```

আউটপুট

```
even
even
odd
```

প্রবলেম - 2

কোনো একটি পূর্ণসংখ্যা দেওয়া থাকলে সেটি জোড় না বিজোড় তা বের করতে হবে।

ইনপুট

প্রথম লাইনে একটি সংখ্যা T ($1 \leq T \leq 100$) দেওয়া থাকবে। পরবর্তীতে T সংখ্যক লাইনে একটি করে অঋণাত্মক (Non-negative) পূর্ণসংখ্যা n দেওয়া থাকবে। একটি সংখ্যায় সর্বোচ্চ 100টি অংক (digit) থাকতে পারে।

আউটপুট

প্রতিটি পূর্ণসংখ্যার জন্য, সংখ্যাটি জোড় হলে even আর বিজোড় হলে odd কথাটি প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
100
0
1111
```

আউটপুট

```
even
even
odd
```

প্রবলেম - 3

এমন একটি প্রোগ্রাম লিখতে হবে যেটি 1 থেকে 1000 পর্যন্ত সবগুলো সংখ্যাকে বড় থেকে ছোট ক্রমানুসারে প্রিন্ট করবে।

ইনপুট

প্রোগ্রামটিতে কোন ইনপুট নেই।

আউটপুট

প্রতিটি লাইনে মোট পাঁচটি (5) করে সংখ্যা থাকবে এবং প্রতিটি সংখ্যা একটি '\t'(Tab) ক্যারেকটার দিয়ে আলাদা করা থাকবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

প্রোগ্রামটিতে কোন ইনপুট নেই

আউটপুট

1000	999	998	997	996
995	994	993	992	991
990	989	988	987	986
...
...
20	19	18	17	16
15	14	13	12	11
10	9	8	7	6
5	4	3	2	1

প্রবলেম - 4

(বাংলাদেশ ইনফরমেটিক্স অলিম্পিয়াডের ট্রেনিং প্রোগ্রাম থেকে সংগৃহীত)

একটি সংখ্যার সমস্ত গুণনীয়ক (ভাজক) বের করতে হবে।

ইনপুট

ইনপুট ফাইলের প্রথম লাইনে থাকবে টেস্ট কেসের সংখ্যা T ($T \leq 10$). এরপরের পরবর্তী T সংখ্যক লাইনের প্রতিটিতে একটি করে পূর্ণ সংখ্যা N থাকবে, ($1 \leq N \leq 100000$).

আউটপুট

প্রতিটি কেসের জন্য একটি করে লাইন প্রিন্ট করতে হবে, শুরুতে কেইস নম্বর দিতে হবে। এরপর N এর সকল গুণনীয়ক ছোট থেকে বড় হিসেবে দেখাতে হবে এবং প্রতিটি গুণনীয়ক শুধুমাত্র একবার দেখাতে হবে। গুণনীয়ক গুলোকে শুধুমাত্র একটি স্পেস দিয়ে আলাদা করতে হবে এবং লাইনের শেষে কোনো অতিরিক্ত স্পেস থাকবে না।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
6
15
23

আউটপুট

Case 1: 1 2 3 6
Case 2: 1 3 5 15
Case 3: 1 23

প্রবলেম - 5

তোমার হাতে যথেষ্ট পরিমাণে কাজকর্ম নেই দেখে তোমাকে একটি বাস্তব আঁকার কাজ দেওয়া হলো। আসলে ব্যাপারটি তেমন কিছু কঠিন নয়, তোমাকে বর্গের একটি বাহুর দৈর্ঘ্য বলা হবে আর তুমি চট করে * অক্ষরটি ব্যবহার করে ওই বর্গটি এঁকে ফেলবে। বর্গের কেবল বাহু আঁকলেই হবে না, ভিতরের ঘরগুলোও * অক্ষরটি দিয়ে পূর্ণ করে দিতে হবে। যেহেতু তুমি প্রোগ্রামিং শেখা শুরু করেছ এবং লুপ পর্যন্ত শিখে ফেলেছ, তাই তুমি কাজটি করবে একটি প্রোগ্রাম লিখে।

ইনপুট

প্রথম লাইনে একটি সংখ্যা T ($1 \leq T \leq 25$) থাকবে এবং তারপর T -সংখ্যক লাইন থাকবে। প্রতিটি লাইনে একটি করে সংখ্যা (N) থাকবে যার মান 1 থেকে 80-এর মধ্যে হবে।

আউটপুট

প্রতিটি N -এর জন্য $N \times N$ বর্গ আঁকতে হবে। পুরো বর্গটি * দিয়ে পূর্ণ করে দিতে হবে। প্রতিটি বর্গ একটি ফাঁকা লাইন দিয়ে পৃথক করে দিতে হবে। পৃথক করার কাজে ব্যাতিত অন্য কোথাও অতিরিক্ত ফাঁকা লাইন বা স্পেস রাখা যাবে না।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
1
3
5

আউটপুট

*

প্রবলেম - 6

পাঁচ (5) অঙ্কের একটি সংখ্যার প্রথম এবং শেষ অঙ্কের যোগফল নির্ণয় করার প্রোগ্রাম লিখতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($1 \leq T \leq 10000$), যা মোট কতটি পাঁচ অঙ্কের সংখ্যা ইনপুট নিতে হবে তা নির্দেশ করবে। পরবর্তীতে T সংখ্যক লাইনে একটি করে পূর্ণসংখ্যা N ($10000 \leq N \leq 99999$) ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে N এর প্রথম এবং শেষ অঙ্কের যোগফল প্রিন্ট করতে হবে নিচের ছকে যেভাবে দেখানো আছে ঠিক সেই ফরম্যাটে। প্রতিটি আউটপুটের পরে একটি নিউলাইন প্রিন্ট করতে হবে। আউটপুটে "=" আগের এবং পরের স্পেসটি লক্ষণীয়।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

5
12345
56789
14310
10062
96587

আউটপুট

Sum = 6
Sum = 14
Sum = 1
Sum = 3
Sum = 16

প্রবলেম - 7

একটি লাইনে অনেকগুলো সংখ্যা দেওয়া থাকবে। সংখ্যাগুলোর মধ্যে এক বা একাধিক স্পেস ক্যারেক্টার থাকবে। লাইনে মোট কয়টি সংখ্যা আছে সেটি বের করতে হবে।

ইনপুট

প্রথম লাইনে একটি সংখ্যা T ($1 \leq T \leq 100$) থাকবে। তারপর T -এর মান যত, ততটি লাইন থাকবে। প্রতিটি লাইনে এক বা একাধিক সংখ্যা থাকবে যাদের পরমমান 10000000-এর বেশি হবে না। একটি লাইনের সংখ্যাগুলোর মধ্যে এক বা একাধিক স্পেস ক্যারেক্টার থাকবে। এক লাইনে কমপক্ষে একটি আর সর্বোচ্চ একশটি সংখ্যা থাকবে।

আউটপুট

প্রতিটি লাইনে মোট কয়টি সংখ্যা আছে সেটি প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
4
1 -2 10000 -50 20 7 445
9
-98 876 65
223 9876452 212
```

আউটপুট

```
7
1
3
3
```

প্রবলেম - ৪

তিনটি সংখ্যা দেওয়া থাকবে। এদেরকে ছোট থেকে বড় আকারে প্রিন্ট করতে হবে।

ইনপুট

ইনপুট ফাইলের প্রথম লাইনে থাকবে টেস্ট কেসের সংখ্যা T ($T \leq 100$)। এরপরে প্রতিটি লাইনে তিনটি করে পূর্ণসংখ্যা n_1, n_2, n_3 থাকবে যেগুলো 1000-এর সমান বা ছোট।

আউটপুট

প্রতিটি কেসের জন্য একটি করে কেস নম্বর প্রিন্ট করতে হবে। এরপরে প্রদত্ত তিনটি সংখ্যাকে ছোট থেকে বড় আকারে সাজিয়ে প্রিন্ট করতে হবে। পাশাপাশি দুটি সংখ্যার মাঝে শুধুমাত্র একটি স্পেস প্রিন্ট করতে হবে। নমুনা আউটপুটে আরো বিস্তারিত দেখতে পারো।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
3 2 1
1 2 3
10 5 6
```

আউটপুট

```
Case 1: 1 2 3
Case 2: 1 2 3
Case 3: 5 6 10
```


প্রবলেম - 9

কোন একটি সংখ্যা পূর্ণবর্গ কি না সেটি বের করার প্রোগ্রাম লিখতে হবে।

ইনপুট

ইনপুট ফাইলের প্রথম লাইনে থাকবে টেস্ট কেসের সংখ্যা T ($T \leq 100$). এরপরে T সংখ্যক লাইন থাকবে। প্রতিটি লাইনে থাকবে একটি করে পূর্ণসংখ্যা N ($0 \leq N \leq 2^{31}$)

আউটপুট

প্রোগ্রামটির আউটপুটে N পূর্ণবর্গ সংখ্যা হলে YES প্রিন্ট করতে হবে। অন্যথায় প্রিন্ট করতে হবে NO।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
16
18
196

আউটপুট

YES
NO
YES

প্রবলেম - 10

তুমি গেছ ক্রিকেট খেলা দেখতে। হঠাৎ খেয়াল করলে ইলেকট্রনিক স্কোরবোর্ডে আর সব ঠিক দেখালেও রানরেট ভুল দেখাচ্ছে। ব্যাপারটা ওদের বলতে গিয়ে জানলে যে ওরাও ভুলটা খেয়াল করেছে। কিন্তু অস্ট্রেলিয়া থেকে আমদানি করা স্কোরবোর্ডের প্রোগ্রাম ঠিক করতে সেই দেশের ইঞ্জিনিয়ার আনতে হবে, তার জন্য মন্ত্রনালয় থেকে অনুমতি, অনুদান, সই, সিলছাপ্পর, এটা-সেটা অনেক কিছু লাগবে বলে কাজটা হয়ে উঠছে না। কিন্তু তুমি বুঝতে পেরেছ ব্যাপারটা স্রেফ একটা প্রোগ্রামিং বাগ। দু লাইনের একটা কোড লিখেই ঠিক করে ফেলতে পারবে। সেটা বলতেই, কিছুটা ভেবে টেকনিশিয়ান লোকটা রাজি হয়ে গেল। শর্ত একটাই, তার বসকে কিছু বলা যাবে না। তুমিও তোমার প্রোগ্রামিং স্কিল সত্যিকার একটা কাজে লাগানোর সুযোগ পেয়ে তুমুল উৎসাহে কোড করতে শুরু করে দিলে।

খেলাটা হচ্ছে 50 ওভারের ওয়ানডে ম্যাচ। প্রতিবার যখন ডিস্পেন্সে ওভারপ্রতি বর্তমান রানের হার (current run rate) এবং জেতার জন্য কাঙ্ক্ষিত রানের হার (required run rate) দেখানো হয় তখন প্রতিপক্ষের করা মোট রান, ব্যাটসম্যানদের বর্তমান রান এবং খেলার আর কত বল বাকি আছে তা জানা থাকে।

উল্লেখ্য, ক্রিকেটে 6 বলে 1 ওভার হয় এবং জিততে হলে প্রতিপক্ষের মোট রানের চেয়ে অন্তত 1 রান বেশি করতে হয়।

ইনপুট

প্রথম লাইনে একটি সংখ্যা T ($1 \leq T \leq 100$) থাকবে। ওই সংখ্যার মান যত, এর পরে ততগুলো লাইনে তিনটি করে সংখ্যা থাকবে। প্রথম সংখ্যাটি প্রতিপক্ষের মোট রান, $r1$ ($1 \leq r1 \leq 1000$), দ্বিতীয় সংখ্যাটি ব্যাটসম্যানদের বর্তমান রান, $r2$ ($1 \leq r2 \leq 1000$) এবং তৃতীয় সংখ্যাটি খেলার আর কত বল, B ($1 \leq B \leq 300$) বাকি আছে তা নির্দেশ করে।

আউটপুট

প্রতিলাইনের জন্য সেই লাইনের দেওয়া তথ্য থেকে হিসাব করে প্রথমে ওভারপ্রতি বর্তমান রানের হার এবং এরপর একটি স্পেস দিয়ে কাঙ্ক্ষিত রানের হার প্রিন্ট করতে হবে। প্রতিটি হার অবশ্যই দশমিকের পরে দুই অঙ্ক পর্যন্ত দেখাতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

4

300 294 6

200 100 100

333 250 40

118 100 180

আউটপুট

6.00 7.00

3.00 6.06

5.77 12.60

5.00 0.63

প্রবলেম - 11

এমন একটি প্রোগ্রাম লিখতে হবে যেটি একটি সংখ্যার গৌণিক (মানে, ফ্যাক্টোরিয়াল - factorial) বের করতে পারে।

ইনপুট

প্রথমে একটি সংখ্যা দেওয়া থাকবে এবং ওই সংখ্যার মান যত, তারপর ততটি লাইন থাকবে। প্রতি লাইনে একটি পূর্ণসংখ্যা, N থাকবে। N-এর মান হবে 0 থেকে 15-এর ভেতর ($0 \leq N \leq 15$)।

আউটপুট

প্রতি লাইনে N-এর factorial বা গৌণিক প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
6
10
15

আউটপুট

720
3628800
1307674368000

প্রবলেম - 12

একটি সংখ্যার ফ্যাক্টোরিয়ালের শেষে কতটি শূন্য (0) আছে, তা বের করতে হবে।

ইনপুট

ইনপুটের প্রথম লাইনে থাকবে টেস্ট কেসের সংখ্যা T ($T \leq 100$). এরপরে T সংখ্যক লাইন থাকবে। প্রতিটি লাইনে থাকবে একটি পূর্ণসংখ্যা N ($0 \leq N \leq 10000$)।

আউটপুট

N ফ্যাক্টোরিয়ালের শেষে কতটি শূন্য (0) আছে, তা এক লাইনে প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
6
15
100

আউটপুট

1
3
24

প্রবলেম - 13

টমি মিয়া প্রোবাবিলিটিতে (সম্ভাব্যতা) বিশ্বাস করে।

তো কেউই খুব অবাক হলো না যখন টমি মিয়া তার বড় ছেলেকে ইংলিশ মিডিয়াম স্কুলে ভর্তি করানোর জন্য প্রিপারেশন নেয়া শুরু করলো। এখন সমস্যা হচ্ছে, ভর্তি পরীক্ষায় ইংরেজি অনুবাদ করতে হবে। টমি মিয়ার ছেলে শব্দগুলোর ইংরেজি অনুবাদ জানে, কিন্তু সে ব্যাকরণ জানে না। এখন সমস্যা হচ্ছে ব্যাকরণ না মানলে ঠিক বাক্য গঠন হয় না। যেমন ধরো, 'তুমি ভাত খাও' এটা তুমি যদি ইংরেজিতে অনুবাদ করে লিখো, 'rice eat you' তাহলে কেউ ভাববে না তুমি কবি। সবাই ভাববে তুমি ব্যাকরণ জানো না, কিংবা ভাববে তুমি চাচ্ছো ভাত তোমাকে খেয়ে ফেলুক!

টমি মিয়া প্রোবাবিলিটিতে বিশ্বাস করে। তো প্রতিটা বাক্যের জন্য টমি মিয়া জানতে চায় তার বড় ছেলের সঠিক হবার প্রোবাবিলিটি কতো।

ইনপুট

প্রথম লাইনে একটি পূর্ণসংখ্যা T ($1 \leq T \leq 100$) থাকবে। পরবর্তীতে T সংখ্যক লাইন থাকবে এবং প্রতি লাইনে একটি ইংরেজি বাক্য থাকবে - সেখানে ১০টির বেশি শব্দ থাকবে না। প্রতি শব্দে ২০টির বেশি বর্ণ থাকবে না।

আউটপুট

প্রতি লাইনে অনুবাদ সঠিক হওয়ার প্রোবাবিলিটি প্রিন্ট করতে হবে। তুমি ধরে নিতে পারো, প্রতিটি বাক্যের জন্য শুধু একটি মাত্র সঠিক অনুক্রম আছে। অনুক্রম মানে হচ্ছে order। আউটপুট দেখাতে হবে $1/n$ আকারে যেখানে n একটি পূর্ণ সংখ্যা।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
2
eat you rice
no way no good
```

আউটগুট

1/6

1/12

প্রবলেম - 14

এমন একটি প্রোগ্রাম লিখতে হতে যেটি একটি বাক্যে কোন একটি বিশেষ বর্ণ কতবার আছে সেটি প্রিন্ট করবে।

ইনপুট

প্রথম লাইনে একটি পূর্ণসংখ্যা T ($1 \leq T \leq 100$) থাকবে। পরবর্তীতে $2 * T$ সংখ্যক লাইন থাকবে। পরবর্তীতে $2 * T$ লাইনের প্রথমটিতে একটি স্ট্রিং (string) থাকবে যার দৈর্ঘ্য সর্বোচ্চ 10000 এবং দ্বিতীয় লাইনে একটি ক্যারেক্টার (character) থাকবে।

আউটপুট

প্রোগ্রামটির আউটপুটে ক্যারেক্টারটি স্ট্রিংয়ে কতবার আছে সেটি প্রিন্ট করতে হবে নিম্নরূপে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
2
hello world
1
hello world
a
```

আউটপুট

```
Occurrence of 'l' in 'hello world' = 3
'a' is not present
```


প্রবলেম - 15

এমন একটি প্রোগ্রাম লিখতে হতে যেটি একটি বাক্যে সবগুলো বর্ণ (alphabet) কতবার করে এসেছে সেটি প্রিন্ট করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণ সংখ্যা T ($1 \leq T \leq 100$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইন থাকবে। প্রতিটি লাইনে থাকবে একটি স্ট্রিং (string) S (S এর দৈর্ঘ্য 1000 এর বেশি নয়)। স্ট্রিং এর সবগুলো বর্ণই ছোট হাতের অক্ষরে লেখা থাকবে।

আউটপুট

প্রোগ্রামটির আউটপুটে ক্যারেক্টারটি স্ট্রিং S এ কোন বর্ণটি কতবার আছে সেটি বর্ণক্রমানুসারে প্রিন্ট করতে হবে। আউটপুট "=" চিহ্নের আগে ও পরের স্পেসটি লক্ষণীয়।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
hello
baby
programming
```

আউটপুট

```
e = 1
h = 1
l = 2
o = 1
```

```
a = 1
b = 2
y = 1
```

```
a = 1
g = 2
i = 1
m = 2
n = 1
o = 1
p = 1
r = 2
```

প্রবলেম - 16

এমন একটি প্রোগ্রাম লিখতে হবে, যেটি একটি বাক্যের সবগুলো শব্দকে উল্টো করে দেখাবে।

ইনপুট

ইনপুটের প্রথম লাইনে থাকবে টেস্ট কেসের সংখ্যা $T(T \leq 100)$ । এরপরে T সংখ্যক লাইন থাকবে। প্রতিটি লাইনে থাকবে স্ট্রিং S (S এর দৈর্ঘ্য 1000 এর বেশি নয়)। পাশাপাশি দুটি শব্দ একটি স্পেস ক্যারেক্টার দিয়ে আলাদা করা থাকবে।

আউটপুট

প্রোগ্রামটির আউটপুটে S এর প্রত্যেকটি শব্দকে উল্টো করে প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
This is a test
Hello World
don't underestimate the power of a girl
```

আউটপুট

```
sihT si a tset
olleH dlrow
t'ond etamitserednu eht rewop fo a lrig
```

প্রবলেম - 17

এমন একটি প্রোগ্রাম লিখতে হবে, যেটি একটি বাক্যে কতগুলো স্বরবর্ণ বা vowel আছে সেটি গণনা করবে।

ইনপুট

ইনপুটের প্রথম লাইনে থাকবে টেস্ট কেসের সংখ্যা T ($T \leq 100$). এরপরে T সংখ্যক লাইন থাকবে। প্রতিটি লাইনে থাকবে একটি স্ট্রিং s (s এর দৈর্ঘ্য 1000 এর বেশী নয়)।

আউটপুট

প্রোগ্রামটির আউটপুটে স্ট্রিং S -এর মধ্যে স্বরবর্ণের সংখ্যা (Number of vowels) প্রিন্ট করতে হবে। '=' চিহ্নের আগের এবং পরের স্পেস লক্ষণীয়।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
I am a programmer
happy coding
hello world
```

আউটপুট

```
Number of vowels = 6
Number of vowels = 3
Number of vowels = 3
```

প্রবলেম - 18

এমন একটি প্রোগ্রাম লিখতে হবে যেটি একটি বাক্যে অবস্থিত স্বরবর্ণ (vowels) ও ব্যঞ্জনবর্ণ (consonants) গুলোকে আলাদা আলাদা প্রিন্ট করবে।

ইনপুট

ইনপুটের প্রথম লাইনে থাকবে টেস্ট কেসের সংখ্যা T ($T \leq 100$)। এরপরে T সংখ্যক লাইন থাকবে। প্রতিটি লাইনে থাকবে একটি স্ট্রিং s । s -এ একশটির কম ক্যারেক্টার থাকবে। স্ট্রিংয়ে ক্যাপিটাল কিংবা স্মল লেটার, উভয় ধরনের অক্ষর থাকতে পারে।

আউটপুট

প্রোগ্রামটির অউটপুটে স্ট্রিংটিতে অবস্থিত স্বরবর্ণ ও ব্যঞ্জনবর্ণ গুলোকে আলাদা আলাদা ভাবে পরপর প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

2

this is a very easy
it is a rainy Sunday

আউটপুট

iaaeaa
thssvrysy
iaaaiua
tsrnysndy

প্রবলেম - 19

ইনপুট

```
3
Hello World
Dhaka is the capital of Bangladesh
It is the winter of discontent
```

আউটপুট

```
Count = 2
Count = 6
Count = 6
```

প্রবলেম - 20

ইনপুট

2

Hurrah! was heard on all sides

Hello, I'm Brooker and you're watching TV

আউটপুট

Count = 6

Count = 7

প্রবলেম - 21

এমন একটি প্রোগ্রাম লিখো, যেটি একটি শব্দকে উল্টো করে দেখাবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T, যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইনে প্রোগ্রামটির ইনপুট হবে একটি স্ট্রিং S। স্ট্রিং এর সর্বোচ্চ সীমা হবে 1000।

আউটপুট

প্রোগ্রামটির আউটপুটে, ইনপুট স্ট্রিংটি উল্টো করে প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
string
programming
Aruna
```

আউটপুট

```
gnirts
gnimmargorp
anurA
```

প্রবলেম - 22

প্রাইম নাম্বার বা মৌলিক সংখ্যা হচ্ছে 1 থেকে বড় সেইসব সংখ্যা যগুলোকে 1 এবং ঐ সংখ্যাটি ছাড়া অন্য কোন সংখ্যা দ্বারা ভাগ করা যায় না। যেমন: 11, 13, 17 ইত্যাদি। এমন একটি প্রোগ্রাম লিখতে হবে যেটি একটি নির্দিষ্ট সীমা পর্যন্ত কতগুলো প্রাইম নাম্বার বা মৌলিক সংখ্যা আছে সেটি প্রকাশ করবে।

ইনপুট

প্রোগ্রামটির ইনপুট হবে একটি পূর্ণ সংখ্যা T, যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইনে প্রোগ্রামটির ইনপুট হবে দুইটি পূর্ণ সংখ্যা a ও b। a ও b-এর সর্বোচ্চ সীমা হবে 100000।

আউটপুট

প্রোগ্রামটির আউটপুটে a থেকে b পর্যন্ত সবগুলো প্রাইম নাম্বার আছে, সেটি প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
1 5
25 50
12 27

আউটপুট

3
6
4

প্রবলেম - 23

ইংরেজি বর্ণমালা A,B,C,D... এর সাথে আমাদের পরিচয় হয় শৈশবেই। আমরা জানি ইংরেজি বর্ণমালা ২৬টি। তোমার কাজ হচ্ছে A=1, B=2, ... Z=26 বিবেচনা করে এমন একটি প্রোগ্রাম লেখা যেটি একটি বর্ণগুচ্ছকে সাংখ্যিকগুচ্ছে রূপান্তর করবে। যেমন: ABZ=1226।

ইনপুট

প্রোগ্রামটির শুরুতে টেস্ট কেইসের সংখ্যা ইনপুট দেওয়া হবে। তারপরে প্রতি লাইনে ইনপুট হবে একটি স্ট্রিং S (S এর দৈর্ঘ্য 100 এর বেশী নয়)।

আউটপুট

প্রোগ্রামটির আউটপুটে স্ট্রিং S-এর প্রতিটি অক্ষরের সাংখ্যিক মান প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
ABZ
ZYB
CODING

আউটপুট

1226
26252
31549147

প্রবলেম - 24

একটি অ্যারের উপাদানগুলো থেকে একান্তর উপাদান (Alternate elements) বের করার প্রোগ্রাম লিখতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম লাইনে ইনপুট হবে একটি পূর্ণসংখ্যা যা মোট টেস্ট কেইসের সংখ্যা নির্দেশ করে। আর প্রতি টেস্ট কেসে প্রথম লাইনে ইনপুট হবে একটি পূর্ণসংখ্যা n , যা অ্যারের উপাদান সংখ্যা নির্দেশ করে। পরের লাইনে অ্যারের n সংখ্যক উপাদান ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে অ্যারের একান্তর উপাদানগুলো পাশাপাশি প্রিন্ট করতে হবে। প্রতিটি উপাদান একটি স্পেস ক্যারেক্টার দ্বারা আলাদা থাকবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
2
5
1 2 3 4 5
10
1 4 55 66 22 0 76 11 23 78
```

আউটপুট

```
1 3 5
1 55 22 76 23
```

প্রবলেম - 25

দুইটি সংখ্যার ল.সা.গু (LCM) নির্ণয় করার প্রোগ্রাম লিখতে হবে।

ইনপুট

প্রথম লাইনে থাকবে একটি পূর্ণসংখ্যা t , যা টেস্ট কেসের সংখ্যা নির্দেশ করে। t সংখ্যক টেস্ট কেসে প্রতি লাইনে থাকবে দুটি পূর্ণসংখ্যা a ও b ($a, b \leq 100000$)।

আউটপুট

প্রোগ্রামটির আউটপুটে a ও b এর লসাগু প্রিন্ট করতে হবে। আউটপুটে “=” চিহ্নের আগে ও পরের স্পেস ক্যারেক্টার দিতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
2
30 15
12 16
```

আউটপুট

```
LCM = 30
LCM = 48
```

স্যাম্পল #2

ইনপুট

```
3
14 35
4 6
105 15
```

আউটপুট

```
LCM = 70
LCM = 12
LCM = 105
```

প্রবলেম – 26

ক্লাইং সমস্যা করে কিরিমিরি গ্রহ থেকে এলিয়েন গুপী পৃথিবীতে এসেছে। সে সঙ্গে করে নির্দিষ্ট পরিমাণ খাবার নিয়ে এসেছে। গুপী যে গ্রহ থেকে এসেছে সেখানকার প্রাণীরা প্রতিদিনের জন্য সরবরাহ করা খাবারের অর্ধেক একদিনে খেয়ে থাকে। খাবার যখন এক কেজির কম বা সমান হবে গুপী তখন গ্রহে ফিরে যাবে।

তোমার কাজ হচ্ছে এমন একটি প্রোগ্রাম লিখা, যেটি নির্দিষ্ট পরিমাণ খাবার (কেজি-তে) শেষ করতে গুপীর কতদিন লাগবে সেটি বের করা।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি ইন্টিজার নাম্বার t ($1 \leq t \leq 1000$) যেটি টেস্ট কেসের সংখ্যা বুঝায়। প্রতিটি টেস্ট কেসে একটি দশমিক নাম্বার X ইনপুট নিতে হবে যেটি খাবারের পরিমাণ নির্দেশ করে।

আউটপুট

প্রোগ্রামটির আউটপুটে X কেজি খাবার শেষ করতে গুপীর কয়দিন সময় লাগবে সেটি প্রিন্ট করতে হবে “days” হিসেবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
40.0
200.0
300.0

আউটপুট

6 days
8 days
9 days

প্রবলেম - 27

যদি একটি সংখ্যার প্রতিটি অঙ্কে সংখ্যাটির মোট অঙ্কের সমান পাওয়ার দিয়ে বৃদ্ধি করে, পাওয়ার গুলোকে আবার যোগ করে সেই সংখ্যাটি পুনরায় পাওয়া যায়, তবে সংখ্যাটিকে আর্মস্ট্রং সংখ্যা বলে।
সেক্ষেত্রে তিন অঙ্কের একটি সংখ্যার পাওয়ার বা শক্তি হবে 3। যেমনঃ 153 একটি আর্মস্ট্রং সংখ্যা,

কারণ $1^3 + 5^3 + 3^3 = 1 + 125 + 27 = 153$

এমন একটি প্রোগ্রাম লিখতে হবে যেটি তিন অঙ্কের একটি সংখ্যা আর্মস্ট্রং সংখ্যা কি না, তা নির্ণয় করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা t, যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী t সংখ্যক লাইনে ইনপুট হবে আরো t সংখ্যকটি তিন অঙ্কের পূর্ণসংখ্যা n ($100 \leq n \leq 999$).

আউটপুট

প্রোগ্রামটির আউটপুটে যদি n আর্মস্ট্রং সংখ্যা হয় তাহলে প্রিন্ট করবে "n is an armstrong number!" অন্যথায় প্রিন্ট করবে "n is not an armstrong number!"।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
100
153
371

আউটপুট

100 is not an armstrong number!
153 is an armstrong number!
371 is an armstrong number!

প্রবলেম - 28

এমন একটি প্রোগ্রাম লিখতে হবে যেটি একটি অ্যারে কোন ক্রমে সাজানো বা সর্টেড আছে কিনা তা নির্ণয় করবে।

ইনপুট

প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা n ($n \leq 24$), যেটি অ্যারের সাইজ নির্দেশ করে। এর পরের n সংখ্যক লাইনে n সংখ্যক অ্যারের উপাদান ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে অ্যারেটি সাজানো (sorted) কি না সেটি প্রিন্ট করবে। যদি সাজানো হয় তাহলে প্রিন্ট করবে "YES" অন্যথায় "NO"।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

5
1
2
3
4
5

আউটপুট

YES

স্যাম্পল #2

ইনপুট

10

1

2

3

4

5

6

7

9

10

8

আউটপুট

NO

প্রবলেম - 29

এমন একটি প্রোগ্রাম লিখতে হবে যেটি একটি ক্যারেক্টার uppercase, lowercase, digit নাকি special character সেটি প্রকাশ করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T, যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইনে একটি করে ক্যারেক্টার ch ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে ch ক্যারেক্টারটি uppercase বা বড় হাতের বর্ণ হলে "Uppercase Character" প্রিন্ট করবে, ক্যারেক্টারটি lowercase বা ছোটো হাতের বর্ণ হলে "Lowercase Character" প্রিন্ট করবে, অক্ষ বা ডিজিট হলে "Numerical Digit" প্রিন্ট করবে। অন্যথায় প্রিন্ট করবে "Special Character"।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
4
a
A
5
;
```

আউটপুট

```
Lowercase Character
Uppercase Character
Numerical Digit
Special Character
```


প্রবলেম - 30

যোগ্য সংখ্যা বা Perfect Number হচ্ছে সে সকল সংখ্যা, যেসব সংখ্যার ভাজকগুলোর যোগফল উক্ত সংখ্যার সমান। যেমনঃ 6 একটি যোগ্য সংখ্যা, কারণ এটি 1, 2, 3 দ্বারা বিভাজ্য এবং এই তিনটি সংখ্যার যোগফল 6 ($1+2+3=6$)। তোমার কাজ হচ্ছে এমন একটি প্রোগ্রাম লেখা যেটি একটি সংখ্যা পারফেক্ট কি না, সেটি প্রকাশ করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T, যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক পূর্ণসংখ্যা N ($N \leq 2^{64}-1$) ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে N পারফেক্ট নাকি না হলে "YES, N is a perfect number!" অন্যথায় "NO, N is not a perfect number!" প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
6 28 30

আউটপুট

YES, 6 is a perfect number!
YES, 28 is a perfect number!
NO, 30 is not a perfect number!

প্রবলেম - 31

যোগ্য সংখ্যা বা Perfect Number হচ্ছে সে সকল সংখ্যা, যেসব সংখ্যার ভাজকগুলোর যোগফল উক্ত সংখ্যার সমান। যেমনঃ 6 একটি যোগ্য সংখ্যা, কারণ এটি 1, 2, 3 দ্বারা বিভাজ্য এবং এই তিনটি সংখ্যার যোগফল 6 ($1+2+3=6$)। তোমার কাজ হচ্ছে এমন একটি প্রোগ্রাম লেখা যেটি একটি সংখ্যা পারফেক্ট কি না, সেটি প্রকাশ করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 100$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তীতে T সংখ্যক পূর্ণসংখ্যা N ($N \leq 40000000$) ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে 1 থেকে N পর্যন্ত সবগুলো যোগ্য সংখ্যা বা পারফেক্ট নাম্বার প্রিন্ট করতে হবে। দুটো আউটপুট সেটের মধ্যে একটি ফাঁকা লাইন থাকবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

2
100 500

আউটপুট

6
28

6
28
496

প্রবলেম - 32

এমন একটি প্রোগ্রাম লিখতে হবে, যেটি একটি সংখ্যা N পর্যন্ত একটি সংখ্যা X -এর গুণিতকসমূহ প্রিন্ট করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 100$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইনে দুইটি করে পূর্ণসংখ্যা X এবং N ($N \leq 1000000$) ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে N পর্যন্ত X এর গুণিতকসমূহ প্রিন্ট করতে হবে। যদি X এর মান N এর থেকে বড় হয় তাহলে "Invalid!" প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
2 10
99 1000
10 5
```

আউটপুট

```
2
4
6
8
10
```

```
99
198
297
396
495
594
693
792
891
990
```

Invalid!

প্রবলেম - 33

এমন একটি প্রোগ্রাম লিখতে হবে, যেটি একটি নির্দিষ্ট সীমার অন্তর্ভুক্ত যতগুলো সংখ্যা অপর একটি পূর্ণসংখ্যা দ্বারা নিঃশেষে বিভাজ্য সেটি বের করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 100$), যা টেস্ট কেসের সংখ্যা নির্দেশ করবে। পরবর্তী T সংখ্যক লাইনে তিনটি পূর্ণসংখ্যা A, B, C ($1 \leq A, B, C \leq 10^{16}$) ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে A থেকে B পর্যন্ত যতগুলো সংখ্যা C দ্বারা নিঃশেষে বিভাজ্য সেই সংখ্যাগুলো প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
2 20 3
50 60 5
55 100 6
```

আউটপুট

```
3
6
9
12
15
18

50
55
60

60
66
72
78
84
90
96
```

প্রবলেম - 34

এমন একটি প্রোগ্রাম লিখতে হবে, যেটি 1 থেকে শুরু করে একটি নির্দিষ্ট সীমার অন্তর্ভুক্ত যতগুলো সংখ্যা দিয়ে নিঃশেষে বিভাজ্য সেটি বের করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 100$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইনে তিনটি পূর্ণসংখ্যা A, B, C ($1 \leq A, B \leq 10^9$, এবং $C \leq 10^{16}$)।

আউটপুট

প্রোগ্রামটির আউটপুটে 1 থেকে C পর্যন্ত যতগুলো সংখ্যা A এবং B দ্বারা নিঃশেষে বিভাজ্য সেই সংখ্যাগুলো প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
2 3 50
3 5 50
5 6 100
```

আউটপুট

```
6
12
18
24
30
36
42
48

15
30
45

30
60
90
```

প্রবলেম - 35

একটি স্থানাঙ্ক বৃত্তের বাইরে না ভেতরে অবস্থিত, সেটি বের করার প্রোগ্রাম লিখতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T , যা টেস্ট কেসের সংখ্যা নির্দেশ করবে। পরবর্তী T সংখ্যক ইনপুট থাকবে। প্রতিটি ইনপুটের প্রথম লাইনে থাকবে একজোড়া ধনাত্মক বা ঋণাত্মক বা বৃত্তের কেন্দ্রের স্থানাঙ্ক (X_c, Y_c) , এরপর ইনপুট হবে বৃত্তের ব্যাসার্ধ r । পরে ইনপুট নিতে হবে একটি স্থানাঙ্ক (X, Y) ।

আউটপুট

প্রোগ্রামটির আউটপুটে (X, Y) বৃত্তের অভ্যন্তরে থাকলে প্রিন্ট করতে হবে "The point is inside the circle" অন্যথায় প্রিন্ট করতে হবে "The point is not inside the circle"।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
2
1 1
4
10 -14
1 1
8
5 6
```

আউটপুট

```
The point is not inside the circle
The point is inside the circle
```

প্রবলেম - 36

এমন একটি প্রোগ্রাম লিখতে হবে যেটি একটি সংখ্যাকে উল্টোভাবে দেখাবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T , যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইনে একটি করে পূর্ণসংখ্যা N ($N \leq 3 \times 10^{11}$) ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে N কে উল্টোভাবে প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
612
1431056
10041992

আউটপুট

216
6501341
29914001

প্রবলেম - 37

N-সংখ্যক শব্দকে ইংরেজি আক্ষরিক ক্রমানুসারে (Alphabetically) সাজাতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা N ($N \leq 20$), পরবর্তী N সংখ্যক লাইনে ইনপুট হবে একটি স্ট্রিং S।

আউটপুট

প্রোগ্রামটির আউটপুটে N-সংখ্যক স্ট্রিং S কে আক্ষরিক ক্রমানুসারে প্রিন্ট করতে হবে। আউটপুটের শুরুতে এবং শেষে একটি "newline" বা "\n" প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
5
x-ray
apple
cat
bat
house
```

আউটপুট

```
apple
bat
cat
house
x-ray
```


প্রবলেম - 38

ইনপুট

2
3 1
5 2

আউটপুট

1
1 1
1 1 1
1 1
1

2
2 2
2 2 2
2 2 2 2
2 2 2 2 2
2 2 2 2
2 2 2
2 2
2

প্রবলেম - 39

প্যালিনড্রোম হচ্ছে সেসব শব্দ যেটি সোজা বা উল্টো যেভাবে করেই পড়া হোক না কেন একই থাকে। যেমনঃ রমাকান্তকামার, এই শব্দটি ডানদিক বা বামদিক যেকোন দিক থেকেই পড়া হোক না কেন একই থাকবে। তাই "রমাকান্তকামার" প্যালিনড্রোম। এমন আরেকটি শব্দ হচ্ছে "WOW"। তোমার কাজ হচ্ছে, এমন একটি প্রোগ্রাম লেখা যেটি একটি শব্দ প্যালিনড্রোম কি না সেটি শনাক্ত করবে।

ইনপুট

প্রোগ্রামটির প্রথমে একটি পূর্ণসংখ্যা t ইনপুট নিতে হবে এবং পরবর্তী t সংখ্যক লাইনে ইনপুট হবে একটি করে স্ট্রিং।

আউটপুট

প্রোগ্রামটির আউটপুটে স্ট্রিংটি প্যালিনড্রোম কি না সেটি দেখাতে হবে। যদি প্যালিনড্রোম হয় তাহলে প্রিন্ট করতে হবে "Yes! It is Palindrome!" অন্যথায় প্রিন্ট করতে হবে "Sorry! It is not Palindrome!"।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
wow
string
civic
```

আউটপুট

```
Yes! It is Palindrome!
Sorry! It is not Palindrome!
Yes! It is Palindrome!
```

প্রবলেম - 40

$x^0+x^1+x^2+x^3+x^4+\dots+x^k$ ধারার ফলাফল বের করার প্রোগ্রাম লিখতে হবে।

ইনপুট

প্রথম লাইনে একটি সংখ্যা দেওয়া থাকবে। সংখ্যার মান যত হবে ততগুলো লাইনে দুইটি পূর্ণসংখ্যা X, K ($X \leq 10$ এবং $K \leq 6$) দেওয়া হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে উপরোক্ত সিরিজের ফলাফল প্রিন্ট করতে হবে। এখানে '=' এর আগে ও পরের স্পেসটি লক্ষণীয়।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

2
5 2
2 5

আউটপুট

Result = 31
Result = 63

স্যাম্পল #2

ইনপুট

3
2 10
5 10
10 5

আউটপুট

Result = 2047
Result = 12207031
Result = 111111

প্রবলেম - 41

$$\frac{1}{1!} + \frac{2}{2!} + \frac{3}{3!} + \dots + \frac{n}{n!}$$

এই ধারার যোগফল বের করার প্রোগ্রাম লিখতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 100$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে।
পরবর্তী T সংখ্যক লাইনে ইনপুট হবে একটি পূর্ণসংখ্যা n ($n \leq 15$).

আউটপুট

প্রোগ্রামটির আউটপুটে n পর্যন্ত উপরোক্ত ধারাটির যোগফল প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
5
8
10

আউটপুট

2.7083
2.7183
2.7183

প্রবলেম - 42

$$\sum_{k=n}^0 2^k$$

উপরোক্ত সমীকরণটিকে নমুনা আউটপুটের মত করে প্রিন্ট করতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 100$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে।
পরবর্তী T সংখ্যক লাইনে ইনপুট হবে একটি পূর্ণসংখ্যা n ($n \leq 50$)।

আউটপুট

প্রোগ্রামটির আউটপুটে n পর্যন্ত উপরোক্ত সমীকরণটি প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
5
8
10

আউটপুট

$2^5 + 2^4 + 2^3 + 2^2 + 2 + 1$
 $2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2 + 1$
 $2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2 + 1$

প্রবলেম - 43

$p^q \bmod c$

এর মান নির্ণয় করার প্রোগ্রাম লিখতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 100$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে।

পরবর্তী T সংখ্যক লাইনে ইনপুট হবে তিনটি পূর্ণসংখ্যা p, q, c ($p, q, c \leq 100$)।

আউটপুট

প্রোগ্রামটির আউটপুটে Result প্রিন্ট করতে হবে। আউটপুটে "=" চিহ্নের আগের এবং পরের স্পেসটি লক্ষণীয়।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

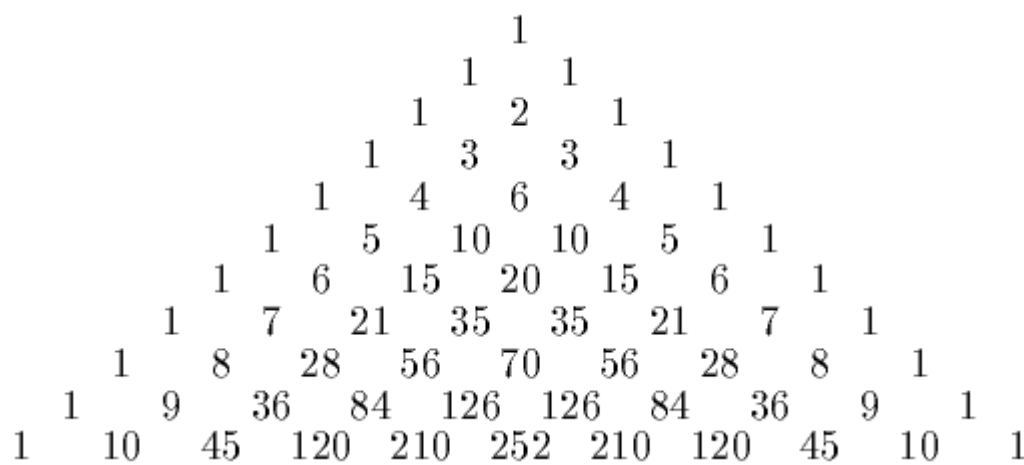
```
3
2 3 3
2 10 5
50 2 3
```

আউটপুট

```
Result = 2
Result = 4
Result = 1
```

প্রবলেম - 44

প্যাসকেলের ত্রিভুজ গণিত শাস্ত্রের একটি আকর্ষণীয় সংখ্যার উদাহরণের নাম। ফারাসি গণিতবিদ এবং দার্শনিক ব্লেইজ প্যাসকেলের নামানুসারে এই সংখ্যা নমুনার (Number pattern) নাম প্যাসকেলের ত্রিভুজ। এই ত্রিভুজের উল্লেখিত যেকোনো সংখ্যা এর ওপরের সারির দুটি সংখ্যার যোগফল। উদাহরণস্বরূপ: পরের ছবির সপ্তম সারির চতুর্থ কলামের সংখ্যাটি 20, যেটি ৬ষ্ঠ সারির ৩য় এবং ৪র্থ কলামের সংখ্যা 10 এবং 10 এর যোগফলের সমান।



তোমার কাজ হচ্ছে এমন একটি প্রোগ্রাম লেখা, যেটি একটি নির্দিষ্ট সারি পর্যন্ত প্যাসকেলের ত্রিভুজ প্রিন্ট করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 20$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইনে একটি করে পূর্ণসংখ্যা N ($N \leq 20$) ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে N তম সারি পর্যন্ত প্যাসকেলের ত্রিভুজ প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

2
3
4

আউটপুট

1

1 1

1 2 1

1 3 3 1

1

1 1

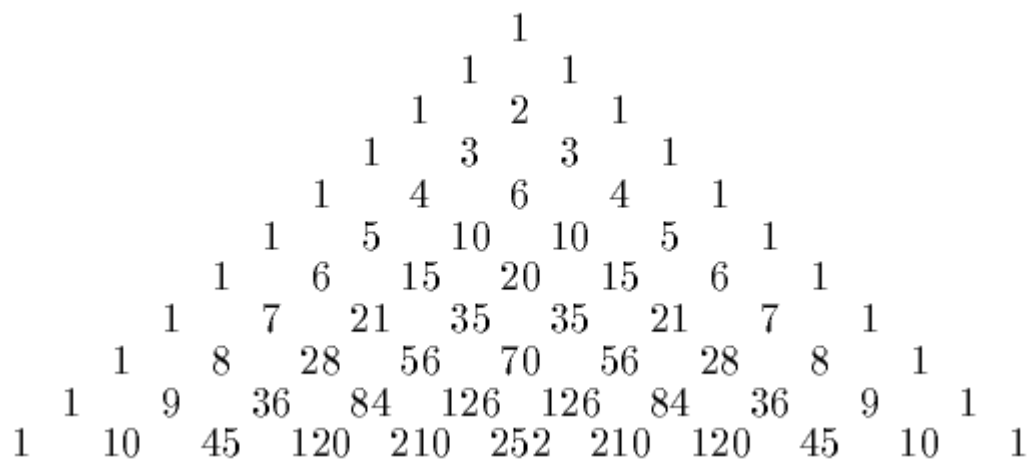
1 2 1

1 3 3 1

1 4 6 4 1

প্রবলেম - 45

প্যাসকেলের ত্রিভুজ গণিত শাস্ত্রের একটি আকর্ষণীয় সংখ্যার উদাহরণের নাম। ফারাসি গণিতবিদ এবং দার্শনিক ব্লেইজ প্যাসকেলের নামানুসারে এই সংখ্যা নমুনার (Number pattern) নাম প্যাসকেলের ত্রিভুজ। এই ত্রিভুজের উল্লেখিত যেকোনো সংখ্যা এর ওপরের সারির দুটি সংখ্যার যোগফল। উদাহরণস্বরূপ: পরের ছবির সপ্তম সারির চতুর্থ কলামের সংখ্যাটি 20, যেটি ৬ষ্ঠ সারির ৩য় এবং ৪র্থ কলামের সংখ্যা 10 এবং 10 এর যোগফলের সমান।



তোমার কাজ হচ্ছে এমন একটি প্রোগ্রাম লেখা, যেটি একটি নির্দিষ্ট সারি পর্যন্ত প্যাসকেলের ত্রিভুজ প্রিন্ট করবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 1000$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে।
পরবর্তী T সংখ্যক লাইনে একটি করে পূর্ণসংখ্যা N ($N \leq 50$) ইনপুট নিতে হবে।

আউটপুট

প্রোগ্রামটির আউটপুটে N তম সারি পর্যন্ত প্যাসকেলের ত্রিভুজ প্রিন্ট করতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

2

3 4

আউটপুট

1

1 1

1 2 1

1 3 3 1

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

প্রবলেম - 46

একটি ত্রিভুজের তিন বাহুর দৈর্ঘ্য দেয়া আছে। ত্রিভুজের ক্ষেত্রফল বের করার প্রোগ্রাম লিখতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 1000$), যা টেস্ট কেসের সংখ্যা নির্দেশ করে। পরবর্তী T সংখ্যক লাইনে থাকবে তিনটি করে পূর্ণসংখ্যা a, b, c । এখানে লক্ষণীয়, ত্রিভুজের যেকোন দুই বাহুর যোগফল অবশ্যই তৃতীয় বাহু অপেক্ষা বড় হবে।

আউটপুট

প্রোগ্রামটির আউটপুট হবে ত্রিভুজের ক্ষেত্রফল Area, যেটি দশমিকের পর ৩ ঘর পর্যন্ত প্রিন্ট করবে। এখানে '=' চিহ্নের আগের ও পরের স্পেসটি লক্ষণীয়।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
3
24 30 18
13 18 15
20 20 20
```

আউটপুট

```
Area = 216.000
Area = 95.917
Area = 173.205
```

প্রবলেম - 47

দুটি ছোট থেকে বড় ক্রমে সাজানো অ্যারে দেওয়া থাকবে। অ্যারে দুটিকে যুক্ত করে একটি ক্রমানুসারে সাজানো অ্যারে তৈরি করতে হবে।

ইনপুট

প্রোগ্রামটির প্রথম ইনপুট হবে একটি পূর্ণসংখ্যা T ($T \leq 1000$), যা টেস্ট কেসের সংখ্যা নির্দেশ করবে। পরবর্তী T সংখ্যক ইনপুট কেস থাকবে। প্রতিটি ইনপুট কেসে দুটি করে লাইন থাকবে। প্রথম লাইনে প্রথম একটি সংখ্যা n_1 থাকবে এবং তারপরে n_1 সংখ্যক পূর্ণসংখ্যা থাকবে। একইভাবে দ্বিতীয় লাইনে প্রথমে একটি সংখ্যা n_2 থাকবে এবং তারপরে n_2 সংখ্যা পূর্ণসংখ্যা থাকবে। দুষ্ক্রেই সংখ্যাগুলো ছোট থেকে বড় ক্রমে সাজানো থাকবে।

আউটপুট

প্রতিটি ইনপুট কেসের জন্য একটি লাইনে $n_1 + n_2$ সংখ্যক সংখ্যা আউটপুট দিতে হবে। যেখানে দুটি অ্যারের সবগুলো সদস্য থাকবে ছোট থেকে বড় ক্রমে সাজানো।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
2
3
1 3 5
2
4 10
5
10 20 30 40 50
3
15 21 22
```

আউটপুট

```
1 3 4 5 10
10 15 20 21 22 30 40 50
```

প্রবলেম - 48

ইনপুট

2

7 2 1 4 6 5 3

9 9 4 5 8 6 1 7 2

আউটপুট

7

3

প্রবলেম - 49

একটি সংখ্যা মৌলিক কিনা বের করতে হবে। মৌলিক সংখ্যা হচ্ছে ১ এর চেয়ে বড় পূর্ণসংখ্যা যা শুধুমাত্র ১ এবং নিজেকে দ্বারা বিভাজ্য।

ইনপুট

ইনপুট ফাইলের প্রথম লাইনে থাকবে টেস্ট কেসের সংখ্যা T ($T \leq 10$), এরপরে T সংখ্যক লাইন থাকবে যাদের প্রতিটিতে একটি করে পূর্ণ সংখ্যা N ($2 \leq N \leq 1000000000000$) থাকবে।

আউটপুট

প্রতিটি টেস্ট কেসের জন্য, যদি N মৌলিক হয়, প্রথমে প্রিন্ট করবে N , তারপরে “ is a prime” স্ট্রিংটি কোনও কোটেশন ছাড়া প্রিন্ট করবে। N মৌলিক না হলে প্রথমে প্রিন্ট করবে N , তারপরে “ is not a prime” স্ট্রিংটি কোনও কোটেশন ছাড়া প্রিন্ট করবে। নমুনা আউটপুটে আরো বিস্তারিত দেখতে পারো।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

3
2
6
11

আউটপুট

2 is a prime
6 is not a prime
11 is a prime

প্রবলেম - 50

একটি স্ট্রিং দেওয়া থাকবে। স্ট্রিংটি কিছু দশমিক ডিজিট এবং L, R এই দুইটি ক্যারেকটার দিয়ে গঠিত। স্ট্রিংটির যেসব অবস্থানে L পাওয়া যাবে সেগুলোকে তার ঠিক বামের ক্যারেকটার দিয়ে বদলে ফেলতে হবে। এবং যেসব স্থানে R পাওয়া যাবে সেগুলোকে বদলে ফেলতে হবে তার ডানের ক্যারেকটার দিয়ে। অর্থাৎ, ইনপুট 34R92L6 থাকলে হয়ে যাবে 3499226।

ইনপুট

প্রথম লাইনে একটি সংখ্যা থাকবে। ঐ সংখ্যার মান যত, এর পরে ততগুলো লাইনে ইনপুট স্ট্রিং থাকবে। স্ট্রিংটি শুরু হবে একটি ডিজিট দিয়ে এবং শেষ হবে একটি ডিজিট দিয়ে। মধ্যবর্তী কোন স্থানে ডিজিট ব্যতীত অন্য কোনো ক্যারেকটার পাশাপাশি থাকবে না। প্রতিটি ইনপুট স্ট্রিং এর দৈর্ঘ 50 বা তার কম হবে।

আউটপুট

প্রতিলাইনের জন্য সেই লাইনে দেওয়া স্ট্রিংটিকে নিয়ম অনুযায়ী পরিবর্তন করলে যে নতুন স্ট্রিং পাওয়া যাবে সেটি প্রিন্ট করাতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

5
0L7
4R5L9
71
8R4R0
34R92L6

আউটপুট

007
45559
71
84400
3499226

প্রবলেম - 51

দুইটি স্ট্রিং দেওয়া থাকবে যার দ্বিতীয়টি প্রথমটির সাবস্ট্রিং। খুঁজে বের করতে হবে প্রথমটিতে সাবস্ট্রিংটি সর্বপ্রথম কোথা থেকে শুরু হয়েছে। উল্লেখ্য, কোনো স্ট্রিং এর একটানা কোনো অংশকে বলে তার সাবস্ট্রিং। যেমন, banana এর একটা সাবস্ট্রিং ana এবং এটা সর্ব প্রথম শুরু হয়েছে 1 তম স্থান থেকে। আরেকটি সাবস্ট্রিং ban যা শুরু হয়েছে 0 তম স্থান থেকে। ওদিকে anna প্রথম স্ট্রিংটির বৈধ সাবস্ট্রিং নয়।

ইনপুট

প্রথম লাইনে একটি সংখ্যা থাকবে। ওই সংখ্যার মান যত, এর পরে ততগুলো লাইনে দুইটি করে স্ট্রিং থাকবে। প্রতিটি স্ট্রিংএর দৈর্ঘ্য 128 এর কম এবং স্ট্রিং দুটি একটি স্পেস দিয়ে আলাদা।

আউটপুট

প্রতিলাইনের জন্য সেই লাইনের দ্বিতীয় স্ট্রিংটি প্রথম স্ট্রিংতে সর্বপ্রথম কোথা থেকে শুরু হয়েছে তা বলতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
4
banana ana
banana ban
aquickbrownfoxjumpsoverthelazydog fox
foobar foobar
```

আউটপুট

```
1
0
11
0
```


প্রবলেম - 52

দুইটি স্ট্রিং দেওয়া থাকবে যার দ্বিতীয়টি প্রথমটির সাবস্ট্রিং। খুঁজে বের করতে হবে প্রথমটিতে সাবস্ট্রিংটি কতবার আছে। উল্লেখ্য, কোনো স্ট্রিং এর একটানা কোনো অংশকে বলে তার সাবস্ট্রিং। যেমন, banana এর একটা সাবস্ট্রিং ana এবং এটা সর্ব প্রথম শুরু হয়েছে 1-তম স্থান থেকে, আবার 3-তম স্থান থেকেও এটি আরেকবার আছে।

```
banana
| | | |
ana | |
| | |
ana
```

অর্থাৎ, মোট 2 বার সাবস্ট্রিংটিকে পাওয়া যাচ্ছে। ওদিকে anna প্রথম শব্দটির কোনো বৈধ সাবস্ট্রিংই নয়। তাই পাওয়া যাচ্ছে 0 বার।

ইনপুট

প্রথম লাইনে একটি সংখ্যা থাকবে। ওই সংখ্যার মান যত, এর পরে ততগুলো লাইনে দুইটি করে স্ট্রিং থাকবে। প্রতিটি স্ট্রিংএর দৈর্ঘ্য ১২৮-এর কম এবং স্ট্রিং দুটি একটি স্পেস দিয়ে আলাদা।

আউটপুট

প্রতিলাইনের জন্য সেই লাইনের দ্বিতীয় স্ট্রিংটি প্রথম স্ট্রিংটির মধ্যে কতবার আছে তা বলতে হবে।

স্যাম্পল ইনপুট আউটপুট

স্যাম্পল #1

ইনপুট

```
5
banana ana
banana anna
fox aquickbrownfoxjumpsoverthelazydog
ddddd ddd
foobar foobar
```

আউটপুট

```
2
0
0
3
1
```

