**Алим**

Ангараг гаригт n ширхэг алимны моднууд OX тэнхлэгийн дагуу ялгаатай байрлалд ургадаг мөн нэг алимны модноос нэг л алим гардаг гэдгийг эрдэмтэд тогтоожээ. Ийм бол Ангараг гаригт хол орших 2 алимны модны байрлалын абсолют зөрүү нь d - зайнаас хэтрүүлэх үгүйгээр хичнээн янзаар 3 алим түүж болох вэ?

**Input format**

Эхний мөрөнд хоёр тоо буюу **n** болон **d** тоонууд. Дараагийн мөрөнд өсөх дарааллаар алимны моднуудын байрлалуудыг илэрхийлэх ширхэг тоонууд. Дурын 2 алимны модны байрлалын абсолют зөрүү нь d- өөс хэтрэхгүй болно.

**Constraints**

* **1 ≤ n ≤ 1000**
* **1 ≤ d ≤ 10^9**
* Өгөгдөх тэмдэгт мөрүүд жижиг латин үсгүүдээс тогтно.

**Output**

Хариу болох ганц тоог хэвлэнэ үү.

**Sample testcases**

**Sample Input 0**

4 2

1 2 3 4

**Sample Output 0**

2

**Solution**

#include <iostream>

using namespace std;

int main() {

int N, d, ans = 0;

cin >> N >> d;

int apple[N];

for (int i = 0; i < N; i++) {

cin >> apple[i];

}

for (int i = 0; i < N; i++) {

for (int j = i + 2; j < N; j++) {

if (apple[j] - apple[i] <= d) {

ans += j - i - 1;

}

}

}

cout << ans << endl;

return 0;

}

# Энгийн хувиргалт

Өгсөн тоо тэгш байвал 2-т хуваах, сондгой байвал 3-т үржүүлээд 1-г нэмэх үйлдлийг энгийн хувиргалт гэе.

Хэрэв танд $N$ тоо өгөдсөн бол тухайн тоог 1 болтол хэдэн энгийн хувиргалт хийх шаардлагатай вэ?

## Input

$N$ тоо.

## Output

Хариу болох ганц тоо.

## Constraints

Өгөгдсөн тооны харгалзах хариу нь $1000$-с хэтрэхгүй.

##### Sample Input 0

11

##### Sample Output 0

11 > 34 > 17 > 52 > 26 > 13 > 40 > 20 > 10 > 5 > 16 > 8 > 4 > 2 > 1

##### Sample Input 1

1

##### Sample Output 1

0

Өгөгдсөн тоо 1 байсан учир энгийн хувиргалт хийх шаадлагагүй буюу 0 удаа хийнэ.

## Solution

#include <iostream>

using namespace std;

int main() {

int N, ans = 0;

cin >> N;

while (N != 1) {

if (N % 2 == 0) {

N /= 2;

} else {

N = 3 \* N + 1;

}

ans++;

}

cout << ans << endl;

return 0;

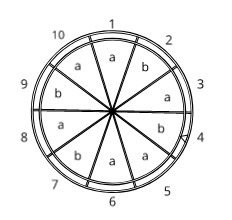
}

**Эртний эдлэл**

Сүүлэнхүү зугаалгаар явж байхдаа агуйгаас эртний эдлэл олжээ. Тэрхүү эртний эдлэлийг ажвал дугуй хэлбэртэй, **N**ширхэг латин үсэг цагиргалан бичигдсэн, гаднаа эргүүлж болох сумтай цагирагтай байв.

Шинжээчид дүгнэлт хийгээд нэгэн **M** урттай тусгай тэмдэгт мөр гарган авчээ. Энэхүү тэмдэгт мөрийн тусламжтайгаар энэхүү эртний эдлэлийн нууцыг тайлж болно гэнэ. Хэрэв эртний эдлэлийн гаднах цагиргийг эргүүлээд тухайн сумны зааж байгаа байрлалаас эхлэн нар зөв **M** ширхэг үсгээр үүссэн тэмдэгт мөр тусгай тэмдэгт мөртэй ижил байвал нууц нь тайлагдах аж.

Ийнхүү Сүүлэнхүү таниас эртний эдлэлийн нууцыг тайлах бүх байрлалуудыг олж өгөхийг хүсчээ.

[](https://camo.githubusercontent.com/2ad27f12338c589b5c4d69d876ba98ce92543f83/68747470733a2f2f73332e616d617a6f6e6177732e636f6d2f68722d6173736574732f302f313534333231303033362d383064343537363533312d44726177696e67312e6a706567)

Эхний жишээ тестийн эртний эдлэлийн гаднах цагиргийн сумыг 4-р байрлалд шилжүүлсэн байдал.

**Input format**

**N** болон **M** урттай тэмдэгт мөрүүд тус бүр шинэ мөрөнд байрлана.

**Constraints**

* **1 ≤ N ≤ 1000**
* **1 ≤ M ≤ 1000**
* **M ≤ N**
* Өгөгдөх тэмдэгт мөрүүд жижиг латин үсгүүдээс тогтно.

**Output**

Хэрэв нууцыг тайлж чадах байрлал олдохгүй бол -1 гэж, хэрэв олдвол хэдэн байрлал байгааг хэвлээд дараагийн мөрөнд тухайн байрлалуудыг **өсөх эрэмбээр** хэвлэ.

**Sample testcases**

**Sample Input 0**

ababaababa

baab

**Sample Output 0**

2

4 9

**Sample Input 1**

nestacademy

proleague

**Sample Output 1**

-1

**Sample Input 2**

aaaa

a

**Sample Output 2**

4

1 2 3 4

**Solution**

#include <iostream>

using namespace std;

int main() {

string N, M;

int answer = 0;

vector<int> pos;

cin >> N >> M;

N += N;

for (int i = 0; i < N.length() / 2; i++) {

string tmp = "";

for (int j = i; j < i + M.length(); j++) {

tmp += N[j];

}

if (tmp == M) {

answer++;

if (i > N.length() / 2) {

pos.push\_back(i - N.length() / 2);

} else {

pos.push\_back(i);

}

}

}

if (answer == 0) {

cout << -1 << endl;

return 0;

}

cout << answer << endl;

for (int i = 0; i < pos.size(); i++) {

cout << pos[i] + 1 << " ";

}

cout << endl;

return 0;

}

# Харгалзах барааны үнэ

Танд дэлгүүрийн бараануудын нэр болон үнэ өгөгджээ. Тэгвэл дэлгүүрийн касс дээр уншсан барааны нэр өгөгдөхөд харгалзах үнийг хэвлэж гаргана уу.

| **№** | **Нэр** | **Үнэ** |
| --- | --- | --- |
| 1 | uhriin\_mah | 7600 |
| 2 | honinii\_mah | 6400 |
| 3 | atar\_talh | 1200 |
| 4 | suut\_boortsog | 2500 |
| 5 | savhan\_boortsog | 3000 |
| 6 | husliin\_undaa | 1500 |
| 7 | bidnii\_undaa | 2200 |
| 8 | oreo\_jignemeg | 1200 |
| 9 | boodol\_bohi | 900 |
| 10 | shirheg\_bohi | 100 |
| 11 | arviin\_chiher | 50 |
| 12 | venus\_zairmag | 600 |

## Input

Касс дээр уншсан барааны нэр.

## Output

Харгалзах барааны үнийг хэвлэ. Хэрэв уншсан бараа жагсаалтанд байхгүй бол "baraa oldsongui." (хашилтгүйгээр) гэж хэвлэнэ үү.

## Solution

#include <iostream>

using namespace std;

int main() {

string name;

cin >> name;

if (name == "uhriin\_mah") {

cout << 7600 << endl;

return 0;

}

if (name == "honinii\_mah") {

cout << 6400 << endl;

return 0;

}

if (name == "atar\_talh") {

cout << 1200 << endl;

return 0;

}

if (name == "suut\_boortsog") {

cout << 2500 << endl;

return 0;

}

if (name == "savhan\_boortsog") {

cout << 3000 << endl;

return 0;

}

if (name == "husliin\_undaa") {

cout << 1500 << endl;

return 0;

}

if (name == "bidnii\_undaa") {

cout << 2200 << endl;

return 0;

}

if (name == "oreo\_jignemeg") {

cout << 1200 << endl;

return 0;

}

if (name == "boodol\_bohi") {

cout << 900 << endl;

return 0;

}

if (name == "shirheg\_bohi") {

cout << 100 << endl;

return 0;

}

if (name == "arviin\_chiher") {

cout << 50 << endl;

return 0;

}

if (name == "venus\_zairmag") {

cout << 600 << endl;

return 0;

}

cout << "baraa oldsongui." << endl;

return 0;

}

**Picasso and Christmas Tree**

* болон - тэмдэгт ашиглан өндөртэй дараах хэлбэрийн гацуур мод хэвлэ.

**Input**

Ердөө л ***N*** тоо.

**Output**

***N*** мөрөөс тогтох дараах хэлбэрийн гацуур мод.

**Constraints**

* **N ≤ 127**

**Sample Input 0**

1

**Sample Output 0**

\*

**Sample Input 1**

2

**Sample Output 1**

-\*-

\*\*\*

**Sample Input 2**

4

**Sample Output 2**

---\*---

--\*\*\*--

-\*\*\*\*\*-

\*\*\*\*\*\*\*

**Sample Input 3**

8

**Sample Output 3**

-------\*-------

------\*\*\*------

-----\*\*\*\*\*-----

----\*\*\*\*\*\*\*----

---\*\*\*\*\*\*\*\*\*---

--\*\*\*\*\*\*\*\*\*\*\*--

-\*\*\*\*\*\*\*\*\*\*\*\*\*-

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Solution**

#include <iostream>

using namespace std;

void print(char c, int times) {

while (times--) {

cout << c;

}

}

int main() {

int N;

cin >> N;

for (int i = 0; i < N; i++) {

print('-', N - i - 1);

print('\*', 2 \* i + 1);

print('-', N - i - 1);

cout << endl;

}

return 0;

}

**Зөрүү**

Танд **N** x **M** хүснэгт өгөгдөв. Таны даалгавар бол хүснэгтийн мөр бүрийн хувьд тухайн мөрөндөх хамгийн их болон хамгийн бага тоонуудын зөрүүг олох юм.

**Input**

Эхний мөрөнд **N, M** тоонууд байрлана. Дараагийн **N** мөр болгонд **M** тоонууд байрлана.

**Output**

**N** мөр тус бүрд харгалзах хариуг хэвлэнэ үү.

**Constraints**

* \*\* 1 ≤ N, M ≤ 1000\*\*
* Өгөгдөх тоонууд **100000**-с бага байна.

**Sample Input**

3 4

1 1 1 1

9 2 5 4

6 6 3 6

**Sample Output**

0

7

3

**Explanation**

Эхний мөрний хамгийн их нь 1, хамгийн бага нь 1 учир хариу нь **1-1=0** Хоёрдах мөрний хамгийн их нь 9, хамгийн бага нь 2 учир хариу нь **9-2=7** Гуравдах мөрний хамгийн их нь 6, хамгийн бага нь 3 учир хариу нь **6-3=3**

**Solution**

#include <iostream>

using namespace std;

int max(int a, int b) {

return a > b ? a : b;

}

int min(int a, int b) {

return a > b ? b : a;

}

int main() {

int N, M;

cin >> N >> M;

for (int i = 0; i < N; i++) {

int MAX, MIN, tmp;

cin >> tmp;

MAX = MIN = tmp;

for (int j = 1; j < M; j++) {

cin >> tmp;

MAX = max(MAX, tmp);

MIN = min(MIN, tmp);

}

cout << MAX - MIN << endl;

}

return 0;

}