НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача А. Ски лагер

Във фермата на Бобо има N (1 ≤ N ≤ 1000) хълма, където през зимата той организира тренировъчен ски лагер.

Тази зима се оказва, че трябва да се плаща нов данък за ски лагерите, ако разликата във височините между най-високия и най-ниския хълм е повече от 17. Затова Бобо решил да намали височината на някои от най-високите хълмове, като изкопаната маса използва за издигане на някои от най-ниските хълмове. Промяната на височината на хълм с x единици струва x*x лева.

Каква е минималната сума, която ще се плати за промяна височините на хълмовете, така че разликата между най-високия и най-ниския хълм да е най-много 17, ако всяка една промяна е с целочислен брой единици?

Вход

За всеки тест на първия ред е зададен броят на хълмовете. Всеки от следващите редове съдържат височината на един хълм - цяло число от интервала от 0 до 100. Край на входа е -1.

Изход

За всеки тест на отделен ред на стандартния изход да се изведе минималната сума, която Бобо трябва да плати за преустройство на хълмовете, така че разликата между най-високия и най-ниския хълм да е най-много 17 единици.

Вход	Изход
5 20 4 1 24 21 -1	18

Пояснение на примера: *Във фермата има 5 хълма, с височини 1, 4, 20, 21 и 24. Хълмовете с височини 4, 20 и 21 остават без промяна. Хълмът с височина 24 се намалява на височина 21. Това струва 3*3=9 лева. Хълмът с височина 1 се увеличава до височина 4, което струва също 3*3=9 лева.*

НБУ
НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача B. Намаляване на скоростта

Бети се състезава в дисциплината ски бягане на зимните игри. Тя стартира със скорост 1 метър в секунда. След известно време тя се изморява и започва да намалява скоростта: тя се движи със скорост 1/2 метра в секунда след първото забавяне на скоростта, после с 1/3 метра в секунда след второто забавяне и т.н.

Данните кога или къде се случва намаляването на скоростта образуват последователност от събития.

Събитие от вида T 17 означава, че Бети намалява скоростта в даден момент - в примера 17 секунди след началото на състезанието. Събитие от вида D 10 означава, че Бети намалява скоростта на зададено разстояние от старта - в случая 10 метра.

По даден списък от N събития, пресметнете времето в секунди, за което Бети ще измине 1 километър. Резултатът да се закръгли до най-близкото цяло число (0.5 се закръглява нагоре на 1).

На първия ред на стандартния вход е зададен броя на тестовете. За всеки тест на първия ред е зададено числото N - брой на събитията (1 ≤ N ≤ 10000). Всеки от следващите N реда е във вида "T x" или "D x", означаващ, съответно, събитие по време или събитие по разстояние. И в двата случая x е цяло число, такова, че събитието се случва преди Бети да измине разстоянието от 1 километър. Възможно е няколко събития да се случват едновременно, което води изведнъж до по-голямо забавяне на Бети. Събитията може да са зададени разбъркано.

За всеки тест на отделен ред на стандартния изход да се изведе общото време, необходимо на Бети да измине 1 километър.

Вход	**Изход**
1 2 T 30 D 10	2970

Пояснение към примера: *Бети намалява скоростта при време t = 30 и при разстояние d = 10. Тя изминава първите 10 метра със скорост 1 метър в секунда, което отнема 10 секунди. Тогава скоростта се намалява на 1/2 метра в секунда и за още 20 секунди изминава 10 метра. Общо са изминали 30 секунди и отново има намаляване на скоростта. До края остават още 980 метра, които Бети изминава със скорост 1/3 метра в секунда, което отнема още 980 * 3 = 2940 секунди. Следователно, общото време е 10 + 20 + 2940 = 2970 секунди.*

НБУ НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача C. Балансирани отбори

Дванадесет от състезателите на Бобо ще участват в зимните игри. Всеки състезател има "ниво на подготовка" - цяло число между 1 и 1 милион. Бобо иска да ги раздели на 4 отбора от по 3 състезателя всеки. Нивото на подготовка на един отбор е равно на сбора от нивата на подготовка на състезателите от този отбор. Бобо иска да образува балансирани отбори, в смисъл, че разликата между нивата на подготовка на най-добре подготвения и най-слабо подготвения отбор да бъде минимална. По-точно, той иска да минимизира разликата S - s, където S и s са съответно максималната и минимална стойност на нивото на подготовка на отбор.

Помогнете на Бобо да определи минималната възможна стойност на S - s.

Вход

Всеки тест се състои от 12 реда. Всеки ред съдържа нивото на подготовка на един от състезателите. Края на входа е маркиран с -1.

Изход

За всеки тест на отделен ред на стандартния изход да се изведе минималната възможна стойност на S – s.

Вход	Изход
1 2 3 4 5 6 7 8 9 10 11 12 -1	1

Едно възможно решение е разпределянето по отбори по следния начин: (12,1,7), (9,8,3), (10,5,4) и (11,2,6). Първите два отбора имат ниво на подготовка 20, а другите два - ниво 19.

НБУ
НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача D. Билярд

По време на зимните игри, в свободното си време, Бобо играе билярд на правоъгълна маса с размери **N x M**. Таблицата съдържа **N** реда и **M** колони, както и 4 дупки в които да влезе топчето в 4-те ъгъла на масата. Той удря бялата топка по диагонал на масата започвайки от горен ляв ъгъл. Например при **N = 6** и **M = 5**, траекторията на топката ще минава през следния път: 1, 7, 13, 19, **25**, **29**, 23, 17, **11**, 7, **3**, 9, **15**, 19, 23, **27**, **21**, 17, 13, 9, **5** - където ще влезне в дупка.

1	2	**3**	4	**5**
6	7	8	9	10
11	12	13	14	**15**
16	17	18	19	20
21	22	23	24	**25**
26	**27**	28	**29**	30

Пояснение: При 25 се сменя посоката, заради удар в края на масата и затова отива към 29. Същото се случва и на 29, 11, 3, 15, 27 и 21. От 21 към 5 е последната част на правата. Щом стига 5, влиза в дупката.

Напишете програма, която намира броя на моментите, в които топката мени посоката си понеже се е ударила в края на масата.

Вход

За всеки тестов пример от стандартния вход се въвеждат две цели числа – N и M ($2 \le N, M \le 2000$) - размерите на таблицата.

Изход

За всеки тестов пример на отделен ред на стандартният изход се извежда едно цяло число – броя на моментите, в които правата трябва да смени посоката си или ще влезне в дупка.

Вход	**Изход**
6 5	8

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача E. Охлюв

Охлюв се намира на земята и иска да се изкачи до върха на дърво с височина V. За един ден той може да изкачи A метра нагоре, през нощта обаче (когато спи) се спуска B метра надолу. Да се определи колко дни са необходими на охлюва, за да се изкачи от земята до върха на дървото.

Входът се състои от няколко тестови примера, всеки от тях съдържащ три цели числа: A, B и V ($1 \le B < A \le V \le 1000000000$).

За всеки тестов пример трябва да изведете по едно число – търсения брой дни, на нов ред.

Вход	**Изход**
2 1 5 5 1 6 100 99 100000000	4 2 999999901

НБУ
НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране"

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача F. Ръкуване

Няколко души са седнали около кръгла маса. По колко различни начина всички могат едновременно да се ръкуват, така че да няма пресичане на ръце? Всеки човек се ръкува само с дясната си ръка.

Напишете програма, която въвежда две цели числа *n* и *m*, и извежда търсения брой за *n* души, пресметнат по модул *m*.

Вход

От първия ред на стандартния вход се въвежда броя на тестовите примери. Всеки тест се състои от целите числа *n* и *m*, записани на отделен ред.

Изход

На стандартния изход програмата трябва да изведе за всеки тест на отделен ред търсения брой за *n* души, пресметнат по модул *m*.

Ограничения

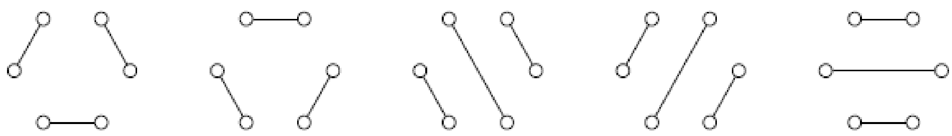
$1 < n < 20\,000$, *n* е четно число

$1 < m < 1001$

Пример

Вход	Изход
1 6 1000	5

Обяснение на примера. Възможните конфигурации при *n* = 6 са следните:



НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача G. Нули

Разглеждаме низ от нули и единици. В него нулите може да се срещат на после-дователни поднизове, например: 00010001011001. В примера има 4 последователности от съ¬сед¬ни нули и това са 000, 000, 0 и 00.

Напишете програма, която въвежда низ от нули и единици и намира колко е броят на различните по дължина низове от последователни нули.

Вход

На първия ред е даден броя на тестовите пример. За всеки тест се следва на отделен ред пое дин низ, съставен само от 0 и 1, и с дължина най-много 10 000 знака.

Изход

За всеки тест да се изведе на отделен ред едно цяло число, равно на търсения брой.

Вход	**Изход**
4 0000 1001001 111 1001000	1 1 0 2

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача H. Максимална *k*-сума

Дадена е редица от n числа a_1, a_2, …, a_n. Да се напише програма, която определя максималната сума на k последователни члена на редицата.

Вход

На първия ред е даден броя на тестовите примери. Следват тестовите примери, като всеки е даден на два реда: от първия се въвеждат числата n и k, а от втория се въвеждат числата от редицата.

Изход

За всеки тест да се изведе на отделен ред търсената максимална сума.

Ограничения

$1 \le k \le n \le 1000$

$-9999 \le a_i \le 9999$, $i = 1, 2, \ldots, n$

Вход	**Изход**
1 6 2 4 -2 3 5 1 3	8

НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача I. Максимуми и минимуми на функция

Дадена е функция f, дефинирана върху множеството от целите положителни числа. Дефинираме минимум на такава функция в точка n, ако f(n) < f(n-1) и f(n) < f(n+1). Аналогично се дефинира и максимум на функцията. Да се намери броят на минимумите и максимумите на такава функция, зададена с редица функционални стойности.

От стандартния вход трябва да се прочетат всички примери. Всеки пример започва с цяло положително число N – броят на точките, където е определена функцията ($N < 10^7$). Следват стойностите на функцията в точките 1, 2, 3, ..., N – цели числа в интервала [-1000, 1000].

За всеки пример на стандартния изход на отделен ред да се изведат две числа – броя на минимумите и броя на максимумите на функцията, отделени с един интервал.

Вход	**Изход**
10 3 2 3 2 2 7 8 2 9 1	2 3

НБУ
НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ
София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика

Школа „Състезателно програмиране”

СЪСТЕЗАНИЕ, 8 март 2014 г.

Задача J. ДВОИЧНО ПРЕДСТАВЯНЕ

Двете устойчиви състояния на електронните цифрови елементи се означават условно с 1 и 0. Те обуславят и основната единица за представяне на информацията в компютъра, наречена бит (от BInary digiT). Понятието разрядност на числата означава броя на цифрите в тях. При записа и обработването им в компютър на всеки разряд на двоичното число съответства двоичен (електронен или магнитен) елемент. Следователно, може да се счита, че информацията се представя в компютъра като съвкупности (масиви, множества) от двоични (дискретни) елементи. Тези масиви образуват т.нар. машинни думи (разрядни решетки). Броят на едновременно обработваните разряди определя разрядността на компютъра и дължината на думата.

Първите компютри са 2-разрядни, 4-разрядни, след това - 8-, 16-, а понастоящем 32- и 64-разрядни. Образуването на двоични числа и операциите с тях са обект на двоичната аритметика, която се основава на двоичната бройна система. Да се намери броя на цифрите със стойност 1 в двоичното представяне в компютъра на цяло положително число.

На стандартния вход е зададена редица от цели числа (по-малки от 10^9, записани в десетична бройна система) която завършва с числото 0. За всеки елемент на редицата (без последния) на стандартния изход се извежда броят на единиците за съответното число от входа - на един ред с разделител интервал.

Вход	**Изход**
1024 99999999 255 0	1 19 8