**Problem A: 是元音字母吗？**

Time Limit: 1 Sec  Memory Limit: 2 MB  
Submit: 10970  Solved: 7019  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4645&pid=0&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1347)]

**Description**

输入一个英文字母，判断是否是元音字母。元音字母是：a,e,i,o u,A,E,I,O,U

**Input**

输入一个英文字母

**Output**

是元音字母,输出“yes”,否则输出“no”,行尾没有回车。

**Sample Input**

A

**Sample Output**

yes

**HINT**

**Problem B: 自增自减运算**

Time Limit: 1 Sec  Memory Limit: 2 MB  
Submit: 14507  Solved: 7111  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4645&pid=1&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1252)]

**Description**

C语言中有自增运算++、自减运算--，并且可以前置和后置。

编写一个程序：对输入的一个整数a，输出a++、a--、++a、--a的值。

**Input**

一个整数a，a是很小的整数。

**Output**

输出为5行，第一行为a的值，后面依次为--a、a--、a++、++a的值。

5行输出纵向a要对齐，“:”对齐，“:”前后各一个空格，后面的数值不用对齐。每行行尾都有回车。

**Sample Input**

0

**Sample Output**

a : 0

--a : -1

a-- : 0

a++ : 0

++a : 1

**HINT**

**Problem C: 求累加和**

Time Limit: 1 Sec  Memory Limit: 2 MB  
Submit: 15349  Solved: 7540  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4645&pid=2&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1145)]

**Description**

编程求min~max的累加和（含min和max），其中max>=min>0。

**Input**

输入为多行。第一行是一个整数N>0，表示后面有N个测试用例。后面有N行，每行包含2个整数，分别是min和max。

**Output**

输出为N行，每个测试用例的计算结果占据一行。每行的格式为：

case i:sum=s.

其中i表示测试用例的编号（从1开始），s是该测试用例对应的累加和（设不超过int的表示范围）。

**Sample Input**

2

1 10

1 100

**Sample Output**

case 1:sum=55.

case 2:sum=5050.

**HINT**

**Problem D: 产生等差序列 之一**

Time Limit: 1 Sec  Memory Limit: 2 MB  
Submit: 20882  Solved: 8895  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4645&pid=3&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1090)]

**Description**

根据给出的初始数、公差和序列长度求等差序列。

**Input**

输入为一行，格式见sample。其中，start为初始数，step为公差，times为序列长度。满足，times>0，step不为0。

**Output**

把这个等差序列输出在一行里，序列两数之间用一个空格分隔。

**Sample Input**

start = 1, step = 2, times = 100

**Sample Output**

1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51 53 55 57 59 61 63 65 67 69 71 73 75 77 79 81 83 85 87 89 91 93 95 97 99 101 103 105 107 109 111 113 115 117 119 121 123 125 127 129 131 133 135 137 139 141 143 145 147 149 151 153 155 157 159 161 163 165 167 169 171 173 175 177 179 181 183 185 187 189 191 193 195 197 199

**HINT**