Roy and his profile picture:

- Roy wants to change his profile picture on Facebook. Now Facebook has some restriction over the dimension of picture that we can upload.
- Minimum dimension of the picture can be L x L, where L is the length of the side of square.
- Now Roy has N photos of various dimensions.
- Dimension of a photo is denoted as W x H
- where W - width of the photo and H - Height of the photo
- When any photo is uploaded following events may occur:
- [1] If any of the width or height is less than L, user is prompted to upload another one. Print "UPLOAD ANOTHER" in this case.
- [2] If width and height, both are large enough and
- (a) if the photo is already square then it is accepted. Print "ACCEPTED" in this case.
- (b) else user is prompted to crop it. Print "CROP IT" in this case.
- (quotes are only for clarification)
- Given L, N, W and H as input, print appropriate text as output.
- Input:
- First line contains L.
- Second line contains N, number of photos.
- Following N lines each contains two space separated integers W and H.
- Output:
- Print appropriate text for each photo in a new line.
- Constraints:
- 1 <= L,W,H <= 10000
- 1 <= N <= 1000

In [8]:

```python
def uploadProfile(l,w,h):
    if w<l or h<l:
        print("UPLOAD ANOTHER")
    elif w==h:
        print("ACCEPTED")
    else:
        print("CROP IT")
l=int(input())
n=int(input())
for i in range(1,n+1):
    s=input().split()
    w=int(s[0])
    h=int(s[1])
    uploadProfile(l,w,h)
```

```
180
3
640 480
CROP IT
120 300
UPLOAD ANOTHER
180 180
ACCEPTED
```

```
In [1]: ord("A")

Out[1]: 65

In [2]: ord("E")

Out[2]: 69

In [3]: ord("I")

Out[3]: 73

In [4]: ord("O")

Out[4]: 79

In [5]: ord("U")

Out[5]: 85

In [6]: ord("Y")

Out[6]: 89

In [ ]: 

In [ ]: 
```

Ali and Helping innocent people:

- Arpasland has surrounded by attackers. A truck enters the city. The driver claims the load is food and medicine from Iranians. Ali is one of the soldiers in Arpasland. He doubts about the truck, maybe it's from the siege. He knows that a tag is valid if the sum of every two consecutive digits of it is even and its letter is not a vowel. Determine if the tag of the truck is valid or not.
- We consider the letters "A","E","I","O","U","Y" to be vowels for this problem.
- Input Format
- The first line contains a string of length 9. The format is "DDXDDD-DD", where D stands for a digit (non zero) and X is an uppercase english letter.
- Output Format
- Print "valid" (without quotes) if the tag is valid, print "invalid" otherwise (without quotes)

In [7]:

```
s=input()
c=8
for i in range(0,len(s)-1,1):
    if s[i].isdigit()==True and s[i+1].isdigit()==True:
        if (int(s[i])+int(s[i+1]))%2!=0:
            print("invalid")
            c=0
            break
    elif (s[i].isalpha()==True) and s[i]!="X" and s[i]!="B" :
        if s[i]=="A" or "E" or "I" or "O" or "U" or "a" or "e" or "i" or "o"
            c=0
            print("invalid")
            break
if(c!=0):
    print("valid")

```

```
12X345-67
invalid
```

Find Product

- You have been given an array A of size N consisting of positive integers. You need to find and print the product of all the number in this array Modulo 10^9+7.
- Input Format:
- The first line contains a single integer N denoting the size of the array. The next line contains N space separated integers denoting the elements of the array
- Output Format:
- Print a single integer denoting the product of all the elements of the array Modulo 10^9+7.

In [9]:

```
N = int(input())
A = input()
p = 1
e = 1000000007
A = A.split(" ")
for i in A: p = (p * int(i)) % e
print(p)
```

```
1000
246 667 10 462 11 119 977 817 882 599 119 908 866 109 870 18 910 169 604 766 95
5 393 522 434 39 379 510 404 649 615 699 574 954 898 588 574 767 459 127 398 91
2 25 82 602 335 711 86 162 332 328 644 999 296 882 200 844 297 264 650 221 246
761 46 756 143 593 510 900 634 115 618 180 777 420 614 41 863 946 961 999 667 4
70 224 107 211 80 866 759 791 903 1000 413 910 292 716 432 706 161 461 494 554
17 904 231 101 977 64 206 599 454 573 432 134 356 447 141 50 228 807 806 847 83
5 104 715 95 163 750 118 639 630 374 609 213 950 68 449 987 129 168 273 842 306
452 475 572 334 284 91 819 154 827 263 455 489 801 831 28 99 300 923 166 28 370
802 948 987 304 825 645 250 317 875 232 595 788 199 625 405 865 513 627 867 98
639 735 514 460 681 727 674 767 987 796 134 961 937 999 267 272 597 680 20 294
170 907 539 444 583 258 866 457 753 384 473 253 291 722 478 264 471 752 436 279
132 143 491 139 20 146 528 182 411 591 226 546 651 909 807 770 903 639 801 892
272 125 190 249 497 682 143 847 478 243 362 661 749 140 811 23 666 428 265 733
507 268 896 486 126 928 352 84 412 533 614 452 728 396 413 703 580 983 83 502 9
63 434 749 285 680 407 129 29 463 265 919 533 980 764 744 493 644 720 100 646 1
934 496 752 464 153 410 648 212 987 228 165 465 920 621 318 526 767 171 576 87
966 344 973 427 216 534 90 81 493 316 884 210 577 630 548 632 166 748 420 124 1
34 537 63 380 190 34 641 133 574 890 790 598 62 771 736 539 150 881 54 354 850
869 416 105 280 442 52 519 786 511 998 269 194 754 745 378 670 832 853 752 806
855 199 181 194 370 86 684 763 914 439 158 440 412 411 989 299 14 200 461 545 4
21 401 128 452 519 8 478 570 693 892 631 940 675 292 820 737 629 89 59 288 859
831 605 40 876 900 268 906 592 248 424 133 45 695 761 993 727 682 417 412 757 1
55 434 551 95 580 846 386 499 566 623 432 378 79 593 671 297 624 647 846 3 719
221 122 396 571 729 25 760 967 32 522 459 107 293 112 945 559 531 979 321 698 1
24 384 176 726 14 467 653 425 730 1 424 803 901 446 465 411 761 742 857 747 273
28 590 257 169 276 791 126 888 124 79 131 665 224 43 1 388 543 672 881 277 273
596 367 514 2 26 380 796 239 164 424 199 716 104 283 872 921 943 624 628 917 55
6 29 33 538 89 951 853 661 292 605 3 314 908 120 985 735 150 273 498 698 900 96
0 466 479 349 548 736 986 614 496 934 473 75 23 594 557 113 524 526 277 454 796
638 417 425 13 750 900 862 132 190 276 317 224 624 574 636 199 17 43 679 312 52
0 207 370 750 767 373 491 402 327 828 181 732 510 882 792 93 296 326 802 218 19
5 591 406 46 466 954 804 111 81 598 791 276 941 81 510 276 513 128 268 654 280
822 615 215 197 458 23 540 352 368 749 707 654 433 391 483 115 204 547 924 508
586 495 602 400 977 795 882 767 432 489 734 448 373 22 987 834 43 602 759 544 7
43 39 174 760 347 544 863 707 305 357 246 866 992 169 549 277 962 879 912 520 9
92 306 286 501 590 192 15 73 408 788 694 666 223 860 580 1 650 494 485 302 519
681 681 719 561 497 813 116 520 810 710 345 5 88 465 408 18 381 718 72 655 36 3
44 378 7 82 971 642 918 714 525 225 493 254 525 454 899 802 482 984 188 586 162
928 894 858 459 893 172 843 753 46 834 679 552 317 508 351 628 691 641 66 877 2
55 149 449 848 504 120 911 761 824 207 695 245 360 805 441 692 102 137 817 988
29 2 210 220 597 828 775 34 465 397 806 453 992 366 501 177 649 689 254 109 220
665 530 98 929 180 675 895 157 172 376 721 796 793 328 574 827 85 770 431 928 7
89 658 985 44 489 252 36 237 279 633 837 98 807 292 842 473 126 888 196 182 454
266 211 450 434 23 882 443 536 862 386 472 872 885 174 247 290 606 74 461 89 68
6 27 713 867 426 320 745 404 549 723 214 792 114 761 813 408 642 829 893 515 88
0 95 487 455 963 504 6 812 277 814 922 67 490 578 784 449 942 41 934 495 191 14
6 744 221 338 391 14 696 400 617 507 588 838 746 71 959 465 122 12 829 871 977
220 276 22 835 854 882 271 79 742 231 21 338 627 960 567 972 177 13 84 156 437
```

```
350 710 524 283 505 61 207 660 662 945 293 598 860 617 719 502 680 794 123 520
328683326
```

In [15]:

```python
def findProduct(n):
    answer=1
    for i in range(1,n+1):
        answer= (answer * i) % (1000000000+7)
    print(answer)

n=int(input())
findProduct(n)
```

```
5
120
```

In [ ]:

```python

```

Aman & Mr.Sharma:

- On a distant planet far away from Earth lives a boy named Aman.He loves to run everyday.But his running distance is directly affected by how much horlicks his mother mixed in his milk today.If his mother has mixed x grams of horlicks,then Aman will be capable of running 100*x meters at most on that day.
- Aman's instructor, Mr.Sharma ,is a very strict yet very caring.Everyday he asks Aman to run around a circle of radius r once.If Aman is able to complete the circle,he would get a toffee.
- Note:Take value of pie=22/7.
- CONSTRAINTS:
- 1<=d<105
- 1<=r<106
- 1<=x<=104
- INPUT:
- First line contains d,no. of days Aman goes to his instructor.Next d lines each contain r,radius of circle and x,amount of horlicks.
- OUTPUT:
- Print total number of toffees Aman would finally have at the end of d days.

In [19]:

```python
d = input()
r = []
j = []
for x in range(int(d)):
    k,l= input().split()

    r.append(k)
    j.append(l)

t=0
for x in range(int(d)):
    distance = 2*int(r[x])*(22/7)
    if int(distance) < 100*int(j[x]):
        t=t+1
print(t)
```

```
3
3 2
4 2
5 1
3
```

In [32]:

```python
s = input()
s=[int(d) for d in str(s)]
ans=0
if(len(s)==10):
    for i in range(0,10):
        ans+=((i+1)*s[i])
        if(ans%11==0):
            print("Legal ISBN")
else:
    print("Illegal ISBN")
if(len(s)!=10):
        print("Illegal ISBN")
```

```
102356412345
Illegal ISBN
Illegal ISBN
```

In [ ]:

```python

```