Amicable numbers

Submission deadline:	**2011-11-06 23:59:59**	533806.245 sec
Evaluation:	**0.0000**	
Max. assessment:	**5.0000** (Without bonus points)	
Submissions:	0 / 10 Free retries + 20 Penalized retries (-2 % penalty each retry)	
Advices:	0 / 2 Advices for free + 2 Advices with a penalty (-10 % penalty each advice)	

The problem is to develop a program which lists amicable numbers.

The input of the program is a positive integer `n` which defines the interval of tested numbers.

The output of the program is a list of amicable numbers. The numbers are listed in an ascending order, one pair per line. The smaller number in the pair is printed first, the greater second. All lines are terminated by a newline.

The program must detect invalid input. If an invalid limit is detected (negative, non-numerical, zero, ...), the program must print an error message and terminate. The error message is terminated by a newline.

Amicable numbers are two different numbers so related that the sum of the proper divisors of each is equal to the other number. For instance numbers `220` and `284` are amicable:

```
divisors of 220: 1, 2, 4, 5, 10, 11, 20, 22, 44, 55, and 110
              1+2+4+5+10+11+20+22+44+55+110 = 284

divisors of 284: 1, 2, 4, 71, and 142
              1+2+4+71+142 = 220
```

The output of your program must exactly match that of the reference. Again, use the enclosed archive and test your program with the provided input/expected output test data (see FAQ). Do not forget newlines, especially after the last line of the output.

Your program will be tested in a restricted environment. The testing environment limits running time and available memory. The exact time and memory limits are shown in the reference solution testing log. The memory limit will hardly cause any problem, however, the time limit is important in this program. A naive solution blindly testing all pairs in the interval will exceed time limit. You need to develop a solution which examines all numbers in the interval only once.

This problem is evaluated in a "bonus" mode. If your program passes all regular tests, it will be awarded nominal points. The optimization mentioned in the previous paragraph is required. To pass the bonus test, you need to develop a faster solution that optimizes the search for divisors.

Please note: the program shall actually find the amicable numbers. It means, that a program that displays (a portion of) a pre-calculated array is not a correct solution. Such programs will be awarded 0 points (we will manually check the programs).

Sample program output:

```
Enter limit:
1000
220, 284
```

```
Enter limit:
10000
220, 284
1184, 1210
2620, 2924
5020, 5564
6232, 6368
```

```
Enter limit:
-5
Invalid input.
```

```
Enter limit:
abcd
Invalid input.
```

Sample data:	**Download**
Submit:	**Submit**

Reference