Q1.

Users are required to enter a side length of a square: x using the keyboard (STDIN).

Please print out the area of the square with 2 decimal places.

Below is an example of how the program will run:

Enter the value 1.77 for 'x'.

```
1.77

OUTPUT:
3.13
Press any key to continue . . .
```

Q2.

Users are required to enter five integer numbers using the keyboard (STDIN).

The program needs to find the maximum even number among the entered values. The program then displays this number on screen.

Below is an example of how the program will run:

```
1
7
2
6
4

OUTPUT:
6
Press any key to continue . . .
```

Q3.

Your program allows users to enter 5 "integer" numbers.

The system sorts the entered numbers in ascending order. The system then displays only the even numbers to screen. There is a newline character between any two adjacent numbers.

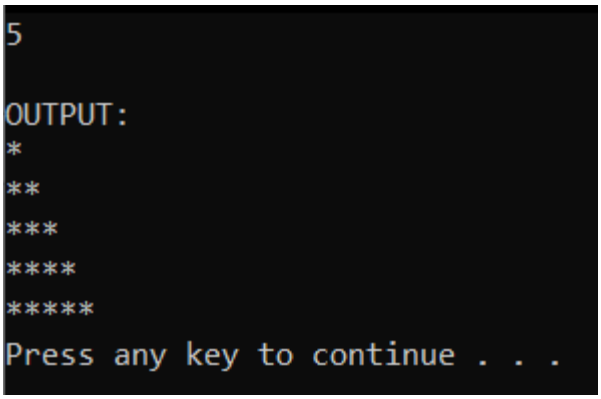
Below is an example of how the program will run:

```
3
2
8
6
7

OUTPUT:
2
6
8
Press any key to continue . . .
```

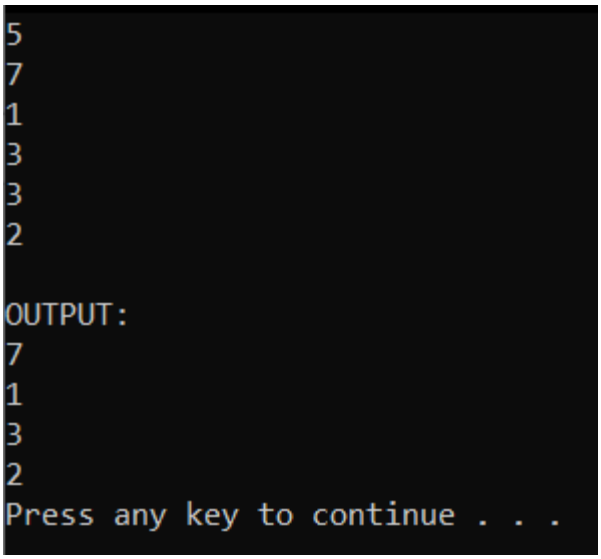
Q4.

Your program allows users to enter height ‘h’ of a pyramid (h < 20).
The program prints out half of the pyramid filled with character ‘*’.
Below is an example of how the program will run:



Q5.

Your program allows users to enter array of n integers, where n is entered by the user (n < 10).
The program removes all duplicated odd numbers (keeps only the first occurrence of the numbers).
Then, the program prints the resultant list of numbers (after removing the duplicated ones). Between any two numbers, there is a newline character.
Below is an example how the program works.



Q6.

Your program allows users to enter a string: ‘s’ with maximum length of 100 characters. The system finds the number of words starting with letter 'h' and ending with letter 'g' in ‘s’. Finally, the system prints out that number.

Below is an example:

s=healing hopping feeling going

```
healing hopping feeling going

OUTPUT:
2
Press any key to continue . . .
```

Q7.

Your program should allow users to enter an array of ‘n’ characters where ‘n’ < 20, ‘n’ is entered by users.

It finds and displays the first two characters appearing the most (having the highest frequencies) among the entered characters.

The program outputs each character on a separate line. The order of output characters follows the order they were entered by users.

Below is the example show how the program works:

```
6
a
a
b
c
e
e

OUTPUT:
a
e
Press any key to continue . . .
```

Q8.

Your program should allow users to enter an integer number: ‘a’. The program should check if ‘a’ is a power of 2 or not. If it is, the program prints the exponent ‘n’ that makes the number ‘a’ the power of 2; else, the program prints: “a is not a power of 2” where ‘a’ is the entered number from user.

Example:

```
256

OUTPUT:
8
Press any key to continue . . .
```

```
255

OUTPUT:
255 is not a power of 2
Press any key to continue . . .
```

Q9.

Your program should allow users to enter a string ‘s’ with maximum 100 characters, then it should display the number of characters in the first three words of ‘s’. Words are separated from each other by a space character.

Examples:

```
hi hello how are you

OUTPUT:
10
Press any key to continue . . .
```

Q10.

Your program should allow users to enter an integer ‘n’.

The program prints hexadecimal representation of ‘n’ if it is a prime number; else the program prints: “n is not a prime number” where ‘n’ is the number entered by the user.

Examples:

```
47

OUTPUT:
0x2F
Press any key to continue . . .
```

```
46

OUTPUT:
46 is not a prime number
Press any key to continue . . .
```