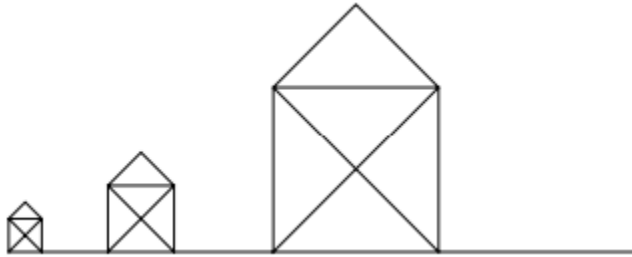
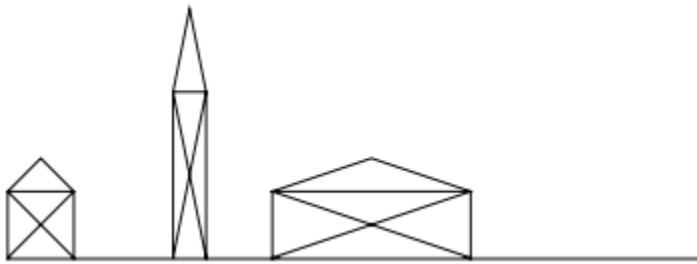


1. What happens when you leave the body of a function empty?
2. What happens if you leave a blank body of a cycle?
3. Write a function that renders the house of a given size.



4. If you like geometry , you can try giving the house function two arguments: width and height. You need to remember the Pythagorean sentence and the tangent function. Attention, the tan function returns the rootstock in radians; you need to convert it on grades (from math import degrees).



5. Change the Stone, Scissors, Paper to repeat the game until the user enters the "end".
6. Change the function yes or no so that instead of "yes" you can use "a" instead of "not" and "n" and not take the letter size and the spaces before / after the answer. Texts such as "maybe" or "sure enough," the computer could not understand it anymore.

7. Write a program that asks the last name of the user / user and tries to guess his / her sex.

8. Find the text of your favorite songs on the Internet, copy it into a string and see how many times it is used letter K.

9. Write the program that simulates this game:

The first player throws a dice (ie random numbers from 1 to 6 are selected) until the sixth falls. Then throws another player,

until it falls to him as well. Then the player throws the third and finally the fourth. The one who wins the sixes wins he needed the most games. (In case of agreement the winner wins the first.)

The program should list all the games and finally write who won

1-D piškvorky se hrají na řádku s dvaceti políčky.
Hráči střídavě přidávají kolečka (o) a křížky (x), třeba:

1. kolo: -----x-----
2. kolo: -----x--o-----
3. kolo: -----xx-o-----
4. kolo: -----xxoo-----
5. kolo: -----xxxoo-----

Hráč, která dá tři své symboly vedle sebe, vyhrál.

10. Write an evaluation function that will get a string with a 1-D gambling array and return a single-string by game status:

"x" - Won the player with crosses (field contains xxx)

"o" - Won the player with wheels (field contains ooo)

"!" - Draw (field not included - and no one wins)

"-" - None of the above situations (i.e., the game has not yet ended)

11. Write a move function that gets a string with a game field, a field number (0-19), and a (x or o) symbol and returns

a game array (i.e., a string) with a given symbol located on the given position.

The function header should look something like this:

def drag (field, symbol, symbol):

"Returns a game field with a given symbol on the given position"

12. Write the tah_hrace function, which gets a string with a

game field, asks the player to which position he wants to play, and returns the game field with the player's recorded turn. The function should reject negative or too large numbers and strokes on occupied boxes. If user he enters the wrong entrance, the function is bouncing and he asks again.

13. Write the `quest_track` that gets the string with the game field, selects the position to play, and returns a game field with recorded computer drag.

Use a simple random "strategy":

1. Select a number from 0 to 19.
2. If the box is empty, play it.
3. If not, repeat from point 1.

The function header should look something like this:

```
def _quot_quocation (field):
```

```
"Returns a game field with recorded computer drag"
```

```
...
```

14. Write the function of the `spline1d`, which creates a string with a game array and alternately calls the `tah_hrace` function and `move_holding` until someone wins or draws.

Do not forget to check the status of the game after every turn

15. Can you program a better strategy for your computer? Maybe to try to play next to his existing ones symbols or to prevent the opponent? Just a small improvement!