



1. ábra. Az első ábra

1. Képek

Ábrák jegyzéke

1.	Az első ábra	1
2.	Ez egy úszó ábra	2
3.	Ez egy transzformált úszó ábra	2
4.	Két részábra egymás mellett.	3

Táblázatok jegyzéke

1.	Egyszerű táblázat.	3
2.	Színes táblázat.	3
3.	Táblázat úszó környezetben, egybevont cellákkal.	4

A kép maradjon úsztatás nélkül, a szöveggel egy sorban. Előtte és utána is generáljuk egy-két bekezdés "zagyva" szöveget:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

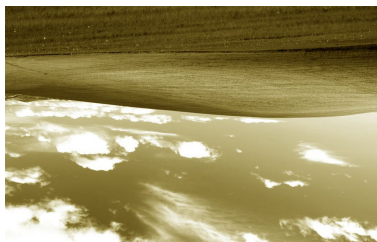


Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Illesszünk be egy másik képet és helyezzük az úszó ábra környezetébe, és pluszban egy másik kép, valamilyen transzformációval:



2. ábra. Ez egy úszó ábra



3. ábra. Ez egy transzformált úszó ábra

Lefordul, de "Hibásan", és +1 számozással mutat egy nem létező ábrára.

Az ábra környezetén belül, hozzunk létre részábra környezetet mindkét képnek:



(a) Első részábra.



(b) Második részábra.

4. ábra. Két részábra egymás mellett.

Első osz- lop	Második oszlop	Harmadik osz- lop
Szöveg 1	Szöveg 2	szöveg 3
Szöveg 7	Szöveg 5	Szöveg 6
		Szöveg 9

1. táblázat. Egyszerű táblázat.

2. Táblázatok

egy	kettő	három
négy	öt	hat
hét	nyolc	kilenc
tíz	tizenegy	tizenkettő

2. táblázat. Színes táblázat.

egy	kettő	
négy	öt	hat
	nyolc	
tíz		

3. táblázat. Táblázat úszó környezetben, egybevont cellákkal.

3. Verbatim

Ez itt egy példa mondat, amiben inline `\textbf{szöveg}` parancsokat használunk.

4. Programkód 1

4.1. python bináris keresés

```
1 def binary_search(arr, val, start, end):
    if start == end:
        if arr[start] > val:
            return start
5         else:
            return start+1
    elif start > end:
        return start
9     else:
        mid = (start+end)/2
        if arr[mid] < val:
            return binary_search(arr, val, mid+1, end)
13        elif arr[mid] > val:
            return binary_search(arr, val, start, mid-1)
        else: # arr[mid] = val
            return mid
17
def insertion_sort(arr):
    for i in xrange(1, len(arr)):
        val = arr[i]
21        j = binary_search(arr, val, 0, i-1)
        arr = arr[:j] + [val] + arr[j:i] + arr[i+1:]
    return arr
```

5. Programkód 2

5.1. c bináris keresés

```
1 binarySearch(arr, x, low, high)
2     repeat till low = high
3         mid = (low + high)/2
4         if (x == arr[mid])
5             return mid
6
7         else if (x > arr[mid]) // x is on the right
            side
8             low = mid + 1
9
10        else // x is on the left side
11            high = mid - 1
```

5.2. c rekurzív bináris keresés

```
1 recursiveBinarySearch(arr, x, low, high)
2     if low > high
3         return False
4
5     else
6         mid = (low + high) / 2
7         if x == arr[mid]
8             return mid
9
10        else if x > arr[mid] // x is on the right side
11            return binarySearch(arr, x, mid + 1, high)
12
13        else // x is on the left side
14            return binarySearch(arr, x, low, mid - 1)
```
