

Aufgabe 1.1

Zeile	i:number	nums:number[]	result:string	Start:string	Length:number	K:number	Kommentar	
2		[2,6,5]					Array nums wird erstellt	
3	0	[2,6,5]					Schleifen beginn	
4	0	[2,6,5]		„m“	2		createCall wird aufgerufen mit Parameter „m“ und 2	
9	0	[2,6,5]		„m“	2	2	K wird mit 2 initialisiert	
10	0	[2,6,5]		„m“	2	2	Bedingung Flash, da k = 2	
12	0	[2,6,5]		„mu“	2	2	An start wird ein „u“ drangehängt	
9	0	[2,6,5]		„mu“	2	1	k wird dekrementiert	
10	0	[2,6,5]		„mu“	2	1	Bedingung true, da k == 1	
11	0	[2,6,5]		„muh“	2	1	Ein „h“ wird drangehängt. An start	
12	0	[2,6,5]		„muhu“	2	1	Ein „u“ wird an start drangehängt	
9	0	[2,6,5]		„muhu“	2	0	Schleife wird verlassen, da k == 0	
14	0	[2,6,5]		„muhu“	2	0	Start wird zurückgegeben	
5	0	[2,6,5]	„muhu“				Auf Konsole wird „muhu“ ausgegeben	
3	1	[2,6,5]	„muhu“				Schleifen beginn, i wird inkrementiert	
4	1	[2,6,5]	„muhu“	„m“	6		createCall wird aufgerufen mit Parameter „m“ und 6	
9	1	[2,6,5]	„muhu“	„m“	6	6	K wird mit 6 initialisiert	
10	1	[2,6,5]	„muhu“	„m“	6	6	Bedingung Falsch, da k = 6	
12	1	[2,6,5]	„muhu“	„mu“	6	6	An start wird ein „u“ drangehängt	
9	1	[2,6,5]	„muhu“	„mu“	6	5	k wird dekrementiert	
10	1	[2,6,5]	„muhu“	„mu“	6	5	Bedingung falsch, da k != 1 && k != 3	
12	1	[2,6,5]	„muhu“	„muu“	6	5	An start wird ein „u“ drangehängt	
9	1	[2,6,5]	„muhu“	„muu“	6	4	k wird dekrementiert	
10	1	[2,6,5]	„muhu“	„muu“	6	4	Bedingung falsch, da k != 1 && k != 3	
12	1	[2,6,5]	„muhu“	„muuu“	6	4	An start wird ein „u“ drangehängt	
9	1	[2,6,5]	„muhu“	„muuu“	6	3	k wird dekrementiert	
10	1	[2,6,5]	„muhu“	„muuu“	6	3	Bedingung true, da k == 3	
11	1	[2,6,5]	„muhu“	„muuuu“	6	3	An start wird ein „h“ drangehängt	
12	1	[2,6,5]	„muhu“	„muuuu“	6	3	An start wird ein „u“ drangehängt	
9	1	[2,6,5]	„muhu“	„muuuu“	6	2	k wird dekrementiert	
10	1	[2,6,5]	„muhu“	„muuuu“	6	2	Bedingung falsch, da k != 1 && k != 3	
12	1	[2,6,5]	„muhu“	„muuuuu“	6	2	An start wird ein „u“ drangehängt	
9	1	[2,6,5]	„muhu“	„muuuuu“	6	1	k wird dekrementiert	
10	1	[2,6,5]	„muhu“	„muuuuu“	6	1	Bedingung true, da k == 1	
11	1	[2,6,5]	„muhu“	„muuuuuu“	6	1	An start wird ein „h“ drangehängt	
12	1	[2,6,5]	„muhu“	„muuuuuu“	6	1	An start wird ein „u“ drangehängt	

Zeile	i:number	nums:number[]	result:string	Start:string	Length:number	K:number	Kommentar	
9	1	[2,6,5]	„muhu“	„muuuuhuhu“	6	0	k wird dekrementiert und schleife verlassen, da k == 0	
14	1	[2,6,5]	„muhu“	„muuuuhuhu“	6	0	start wird zurückgegeben	
5	1	[2,6,5]	„muuuuhuhu“				Auf Konsole wird „muuuuhuhu“ ausgegeben	
3	2	[2,6,5]	„muuuuhuhu“				Schleifen beginn, i wird inkrementiert	
4	2	[2,6,5]	„muuuuhuhu“	„m“	5	5	createCall wird aufgerufen mit Parameter „m“ und 5	
9	2	[2,6,5]	„muuuuhuhu“	„m“	5	5	K wird mit 5 initialisiert	
10	2	[2,6,5]	„muuuuhuhu“	„m“	5	5	Bedingung Falsch, da k = 5	
12	2	[2,6,5]	„muuuuhuhu“	„mu“	5	5	An start wird ein „u“ drangehängt	
9	2	[2,6,5]	„muuuuhuhu“	„mu“	5	4	k wird dekrementiert	
10	2	[2,6,5]	„muuuuhuhu“	„mu“	5	4	Bedingung falsch, da k != 1 && k != 2	
12	2	[2,6,5]	„muuuuhuhu“	„muu“	5	4	An start wird ein „u“ drangehängt	
9	2	[2,6,5]	„muuuuhuhu“	„muu“	5	3	k wird dekrementiert	
10	2	[2,6,5]	„muuuuhuhu“	„muu“	5	3	Bedingung falsch, da k != 1 && k != 2	
12	2	[2,6,5]	„muuuuhuhu“	„muuu“	5	3	An start wird ein „u“ drangehängt	
9	2	[2,6,5]	„muuuuhuhu“	„muuu“	5	2	k wird dekrementiert	
10	2	[2,6,5]	„muuuuhuhu“	„muuu“	5	2	Bedingung falsch, da k != 1 && k != 2.5	
12	2	[2,6,5]	„muuuuhuhu“	„muuuu“	5	2	An start wird ein „u“ drangehängt	
9	2	[2,6,5]	„muuuuhuhu“	„muuuu“	5	1	k wird dekrementiert	
10	2	[2,6,5]	„muuuuhuhu“	„muuuu“	5	1	Bedingung true, da k == 1	
11	2	[2,6,5]	„muuuuhuhu“	„muuuuh“	5	1	An start wird ein „h“ drangehängt	
12	2	[2,6,5]	„muuuuhuhu“	„muuuuhu“	5	1	An start wird ein „u“ drangehängt	
9	2	[2,6,5]	„muuuuhuhu“	„muuuuhu“	5	0	k wird dekrementiert und schleife verlassen, da k == 0	
14	2	[2,6,5]	„muuuuhuhu“	„muuuuhu“	5	0	start wird zurückgegeben	
5	2	[2,6,5]	„muuuuhu“				Auf Konsole wird „muuuuhu“ ausgegeben	
3	3	[2,6,5]	„muuuuhu“				I wird inkrementiert und schleife verlassen, da i == 3	
Ausgabe:								
muhu								
muuuuhuhu								
muuuuhu								

Zeile	i:number	nums:number[]	result:string	Start:string	Length:number	K:number	Kommentar	

Aufgabe 1.2

Zeile	n:number	c:string	X:number	y:number	i:number	a:number	_color:string	_x:number	_y:number	_width:number	_height:number	Kommentar
2	5											Erstelle Variable n und initialisiere 5
3	5											Deklariere variable c
4	5		0									Initialisiere variable x mit 0
5	5		0	0								Initialisiere variable y mit 0
7	5		0	0	0							Initialisiere variable i mit 0 und gehe in die schleife, da 0 < 5
8	5			50	0							Weise y = y + 50 , da i != 2
9	5		170	50	0							Weise x = x + 170, da 170 modulo 400 = 170
10-13	5	„#ff0000“	170	50	0							Gehe zu case 0, da i = 0 und weise c „#ff0000“ zu, verlasse switch
23	5	„#ff0000“	170	50	0	50						Initialisiere variable a mit 50 und gehe in die schleife, da a > 0
24	5	„#ff0000“	170	50	0	50						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv												
25	5	„#ff0000“	170	50		50						Gehe nicht in die bedingung, da i != 4
23	5	„#ff0000“	170	50		30						Ziehe 20 von a ab und gehe in die schleife, da 30 > 0
24	5	„#ff0000“	170	50	0	30						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv												
25	5	„#ff0000“	170	50		50						Gehe nicht in die bedingung, da i != 4
23	5	„#ff0000“	170	50		10						Ziehe 20 von a ab und gehe in die schleife, da 10 > 0
24	5	„#ff0000“	170	50	0	10						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
* PlaceDiv												
25	5	„#ff0000“	170	50		10						Gehe nicht in die bedingung, da i != 4
23	5	„#ff0000“	170	50		-10						Ziehe 20 von a ab und gehe nicht in die schleife, da -10 < 0
7	5	„#ff0000“	170	50	1							Inkrementiere i und gehe in die schleife, da 1 < 5
8	5	„#ff0000“		100	1							Weise y = y + 50 , da i != 2
9	5	„#ff0000“	340	100	1							Weise x = x + 170, da 340 modulo 400 = 340
10,14-17	5	„#00ff00“	340	100	1							Gehe zu case 1, da i = 1 und weise c „#00ff00“ zu, verlasse switch
23	5	„#00ff00“	340	100	1	50						Initialisiere variable a mit 50 und gehe in die schleife, da a > 0
24	5	„#00ff00“	340	100	1	50						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv												
25	5	„#00ff00“	340	100	1	50						Gehe nicht in die bedingung, da i != 4
23	5	„#00ff00“	340	100	1	30						Ziehe 20 von a ab und gehe in die schleife, da 30 > 0
24	5	„#00ff00“	340	100	1	30						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv						30						
25	5	„#00ff00“	340	100	1	30						Gehe nicht in die bedingung, da i != 4
23	5	„#00ff00“	340	100	1	10						Ziehe 20 von a ab und gehe in die schleife, da 10 > 0
24	5	„#00ff00“	340	100	1	10						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv						10						
25	5	„#00ff00“	340	100	1	10						Gehe nicht in die bedingung, da i != 4
23	5	„#00ff00“	340	100	1	-10						Ziehe 20 von a ab und gehe nicht in die schleife, da -10 < 0
7	5	„#00ff00“	340	100	2							Inkrementiere i und gehe in die schleife, da 2 < 5
8	5	„#00ff00“	340	120	2							Weise y = y + 20 , da i == 2
9	5	„#00ff00“	110	120	2							Weise x = 110, da 510 modulo 400 = 110
10,20-21	5	„#0000ff“	110	120	2							Gehe zu case default, da i = 2 und weise c „#0000ff“ zu, verlasse switch
23	5	„#0000ff“	110	120	2	50						Initialisiere variable a mit 50 und gehe in die schleife, da a > 0
24	5	„#0000ff“	110	120	2	50						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv		„#0000ff“	110	120	2	50						

Zeile	n:number	c:string	X:number	y:number	i:number	a:number	_color:string	_x:number	_y:number	_width:number	_height:number	Kommentar
25	5	„#0000ff“	110	120	2	50						Gehe nicht in die bedingung, da i != 4
23	5	„#0000ff“	110	120	2	30						Ziehe 20 von a ab und gehe in die schleife, da 30 > 0
24	5	„#0000ff“	110	120	2	30						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv		„#0000ff“	110	120	2	30						
25	5	„#0000ff“	110	120	2	30						Gehe nicht in die bedingung, da i != 4
23	5	„#0000ff“	110	120	2	10						Ziehe 20 von a ab und gehe in die schleife, da 10 > 0
24	5	„#0000ff“	110	120	2	10						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv			110	120	2	10						
25	5	„#0000ff“	110	120	2	10						Gehe nicht in die bedingung, da i != 4
23	5	„#0000ff“	110	120	2	-10						Ziehe 20 von a ab und gehe nicht in die schleife, da -10 < 0
7	5	„#0000ff“	110	120	3							Inkrementiere i und gehe in die schleife, da 3 < 5
8	5	„#0000ff“	110	170	3							Weise y = y + 50 , da i == 3
9	5	„#0000ff“	280	170	3							Weise x = 280, da 280 modulo 400 = 280
10,18-19	5	„#0000ff“	280	170	3							Gehe zu case 3, da i = 3 und gehe aus switch
23	5	„#0000ff“	280	170	3	50						Initialisiere variable a mit 50 und gehe in die schleife, da a > 0
24	5	„#0000ff“	280	170	3	50						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv			280	170	3							
25	5	„#0000ff“	280	170	3	50						Gehe nicht in die bedingung, da i != 4
23	5	„#0000ff“	280	170	3	30						Ziehe 20 von a ab und gehe in die schleife, da 30 > 0
24	5	„#0000ff“	280	170	3	30						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv			280	170	3	30						
25	5	„#0000ff“	280	170	3	30						Gehe nicht in die bedingung, da i != 4
23	5	„#0000ff“	280	170	3	10						Ziehe 20 von a ab und gehe in die schleife, da 10 > 0
24	5	„#0000ff“	280	170	3	10						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv			280	170	3	10						
25	5	„#0000ff“	280	170	3	10						Gehe nicht in die bedingung, da i != 4
23	5	„#0000ff“	280	170	3	-10						Ziehe 20 von a ab und gehe nicht in die schleife, da -10 < 0
7	5	„#0000ff“	280	170	4							Inkrementiere i und gehe in die schleife, da 4 < 5
8	5	„#0000ff“	280	220	4							Weise y = y + 50 , da i == 4
9	5	„#0000ff“	50	220	4							Weise x = 50, da 450 modulo 400 = 50
10,18-19	5	„#00ff00“	50	220	4							Gehe zu case 4, da i = 4 und weise c = „#00ff00“ zu
23	5	„#00ff00“	50	220	4	50						Initialisiere variable a mit 50 und gehe in die schleife, da a > 0
24	5	„#00ff00“	50	220	4	50						Rufe PlaceDiv auf mit Parameter c,x,y,a,a
*PlaceDiv		„#00ff00“	50	220	4							
25	5	„#00ff00“	50	220	4	50						Gehe nun in die bedingung, da i == 4 und verlasse die Schleife
7	5	„#00ff00“	50	220	5							Inkrementiere i und gehe nicht in die schleife, da i == 5
*PlaceDiv funktion Placeholder												
32												Initialisiere div
33												Rufe Funktion document.body.appendChild auf mit Parameter div
35												Initialisiere variable s:CSSStyleDeclaration mit div.style
36												Weise s.border „thin solid black“ zu
37												Weise s.position „absolute“ zu
38												Weise s.backgroundColor _color zu

Zeile	n:number	c:string	X:number	y:number	i:number	a:number	_color:string	_x:number	_y:number	_width:number	_height:number	Kommentar
39												Weise s.width, den inhalt von _width konkateniert mit „px“ zu
40												Wise s.height den inhalt von _height konkateniert mit „px“ zu
41												Weise s.left, den inhalt von _x konkateniert mit „px“ zu
42												weise s.top, den inhalt von _y konkateniert mit „px“ zu
Die Funktion PlaceDiv wird jeweils mit verschiedenen werten als Parametern aufgerufen, aufgrund Redundanz, habe ich einen Platzhalter benutzt.												

n; number = 5;
c: ~~String~~ String
x: number = 0
y: number = 0

i; number = 0

[i < n]

i++

[i == 2]

y = y + 20

y = y + 50

x = (x + 130) % 400

[0]

[i == 3]

[3]

c = "#ff0000"

[def.]

c = "#0000ff"

c = "#00ff00"

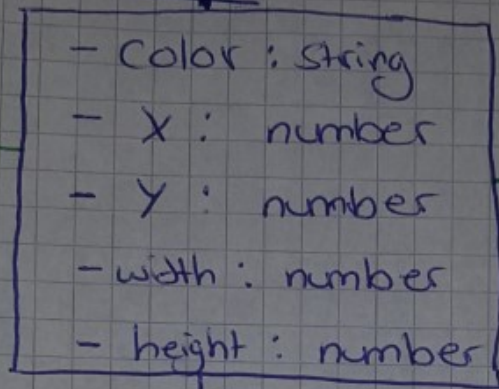
a; number = 50

[a > 0]

PlaceDiv (11)

[i == 4]

a = a - 20



document.createElement

div : HTMLDivElement

document.body.appendChild

s : CSSStyleDeclaration = div.style

s.border = "thin solid black"
s.position = "absolute"
s.backgroundColor = - Color
s.width = - width + "px"
s.height = - height + "px"
s.left = - x + "px"
s.top = - y + "px"

Endknoten

