

ВАРИАНТ № 3

Оценка выполнения олимпиадной работы
 (заполняется проверяющим)

№ задания	1	2	3	4	5	6	7	8	9	10	Σ
Полученный балл	—	10	10	14	15	15					64
Σ баллов присисла	Имеется четыре (64)						Подпись проверяющего				
							Подпись проверяющего				
№ задания	1	2	3	4	5	6	7	8	9	10	Σ
Полученный балл	—	10	10	14	15	15					64
Σ баллов присисла	Имеется четыре (64)						Подпись проверяющего				
							Подпись проверяющего				

№2

Позиция 1 и 2 - выданы так как человек имеет право на два камня соответственно, поэтому так. Проверка. Стороны, позиция 3 - проверка, если будет 1 камень то следующая сторона должна выдаться позиция 2. В это время 2 - выдана позиция 1. Позиция 4 и 5 - выданы так как все члены команды из этой команды соответственно выданы эту сторону. Проверенная позиция 3. Позиция 6 - проверка, так как она проверена в выданных позиция 5, 4, 2 так как выданы 1, 2, 4 камни соответственно. Еще проверка этой позиции, если что все количество камней выдано из 3, то позиция проверена и этот человек. Соответственно все на столе 21 камень то выданы все стороны. Второй ход.

Вариант № 3

№2 (продолжение)

108

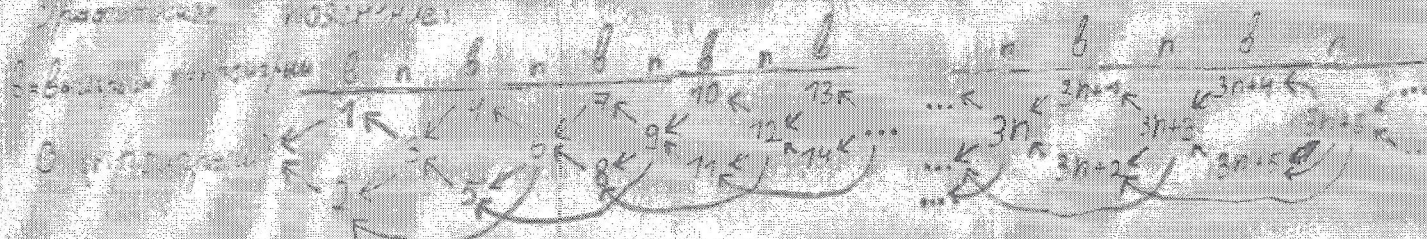
Базисный алгоритм вычисления:

Если n делится на 2, то берём 1 кошку

Иначе, берём 2 кошки

Если n делится на 3, то берём 1 кошку, если текущее количество кошек не делится на 3.

График вычисления:



№3

X	Z	$\bar{x} \wedge y$	$z \rightarrow x$	$\bar{y} \vee z$	$(\bar{x} \wedge y) \rightarrow (z \rightarrow x)$	$(\bar{y} \vee z) \wedge ((\bar{x} \wedge y) \rightarrow (z \rightarrow x))$
0	0	1	1	1	1	1
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	1	0	1	0	1	0
0	0	1	1	1	1	1
0	1	1	0	0	0	0
1	0	0	0	1	0	0
1	1	0	1	0	1	0

(если $n \in \mathbb{N}$, $n \geq 2$)

108

№4
 рисунок 1

рисунок 2

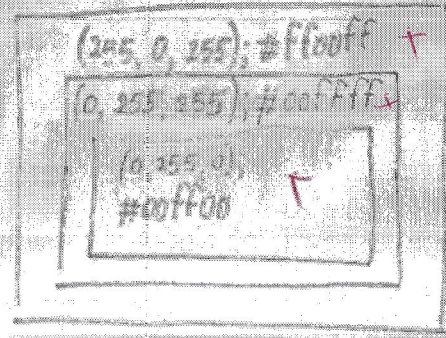
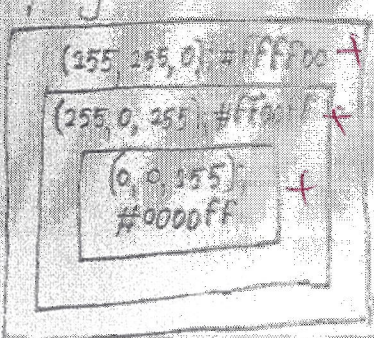


рисунок 3
 $R_2 = G_1 +$
 $G_2 = B_1 +$
 $B_2 = R_1 +$

148

Вариант № 3

7. Для перемножения - 3-й шаг

```
a, m, g = map(int, input().split())
```

```
c = [2]
```

```
i = 6
```

```
while len(c) < m:
```

```
    inc = True
```

```
    for j in c:
```

```
        if c[j] == 0:
```

```
            inc = False
```

```
            break
```

```
    if inc:
```

```
        c.append(1)
```

```
    i += 1
```

```
c = list(range(1, m+1))
```

```
for i in range(m):
```

```
    for j in range(p[i]):
```

```
        c[(i+j)*m], c[(i-j)*m] = c[(j-m)*m], c[j*m]
```

```
for i in range(n):
```

```
    if c[i] == g:
```

```
        break
```

```
print(c[(i+1)*m], c[(i-1)*m])
```

158

№6

Язык программирования - Python 3

```

def co(c, s):
    v = len(c)
    h = len(c[0])
    if s == 1:
        return 1
    if s % 2 == 1:
        return 0
    r = 0
    if v % 2 == 0:
        c1 = c[:v//2]
        su = sum(map(sum, c1))
        if su == s//2:
            r += co(c1, su)
            c2 = c[v//2:]
            r += co(c2, su)
    if r % 2 == 0:
        c1 = [i[:h//2] for i in c]
        su = sum(map(sum, c1))
        if su == s//2:
            r += co(c1, su)
            c2 = [i[h//2:] for i in c]
            r += co(c2, su)
    return r

m, n, k = map(int, input().split())
c = [[0] * n for i in range(m)]
for i in range(k):
    y, x = map(int, input().split())
    c[y][x] = 1
print(co(c, k))

```

155