Группа: Л4−1\_\_, фамилия, имя \_\_\_\_\_\_\_\_\_\_\_\_\_**ОБРАЗЕЦ**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Вариант: 1−1−образец

Выполните трассировку следующих вызовов функций:

1. gcd(84, −66)

2. gcd(0, −41)

3. hex(212)

4. square\_equal(−32, 0, 72)

5. square\_equal(−54, 41, −52)

6. factorize(36)

7. remove\_digit(537, 3)

1. gcd(84, −66)  
1) def gcd(x=84, y=−66)  
2) if 84 < 0: — False  
4) if −66 < 0: — True  
5) y = −−66 = 66  
6) while 66 != 0: — True  
7) rem = 84 % 66 = 18  
8) x = 66  
9) y = 18  
6) while 18 != 0: — True  
7) rem = 66 % 18 = 12  
8) x = 18  
9) y = 12  
6) while 12 != 0: — True  
7) rem = 18 % 12 = 6  
8) x = 12  
9) y = 6  
6) while 6 != 0: — True  
7) rem = 12 % 6 = 0  
8) x = 6  
9) y = 0  
6) while 0 != 0: — False  
10) return 6

2. gcd(0, −41)  
1 def gcd(x=0, y=−41)  
2 if 0 < 0: — False  
4 if −41 < 0: — True  
5 y = −−41 = 41  
6 while 41 != 0: — True  
7 rem = 0 % 41 = 0  
8 x = 41  
9 y = 0  
6 while 0 != 0: — False  
10 return 41

3. hex(212)  
1) def hex(number=212)  
2) if 212 == 0: — False  
4) res = ''  
5) while 212 > 0: — True  
6) digit = 212 % 16 = 4  
7) if 4 <= 9: — True  
8) digit = str(4) = '4'  
23) res = '4' + '' = '4'  
24) number = 212 // 16 = 13  
5) while 13 > 0: — True  
6) digit = 13 % 16 = 13  
7) if 13 <= 9: — False  
9) elif 13 <= 13: — True  
10) if 13 <= 11: — False  
15) elif 13 == 12: — False  
17) else:  
18) digit = 'D'  
23) res = 'D' + '4' = 'D4'  
24) number = 13 // 16 = 0  
5) while 0 > 0: — False  
25) return 'D4'

4. square\_equal(−32, 0, 72)  
3) def square\_equal(a=−32, b=0, c=72)  
4) if −32 != 0: — True  
5) D = 0\*0 − 4\*−32\*72 = 9216  
6) if 9216 > 0: — True  
7) x1 = (−0−sqrt(9216))/(2\*−32) = 1.5  
8) x2 = (−0+sqrt(9216))/(2\*−32) = −1.5  
9) return str(1.5) + ' and ' + str(−1.5)  
 return '1.5 and −1.5'

5. square\_equal(−54, 41, −52)  
3) def square\_equal(a=−54, b=41, c=−52)  
4) if −54 != 0: — True  
5) D = 41\*41 − 4\*−54\*−52 = −9551  
6) if −9551 > 0: — False  
10) elif −9551 == 0: — False  
12) else:  
13) return 'no roots'

6. factorize(36)  
1) def factorize(n=36)  
2) res = ''  
3) while 36 > 2 and 36 % 2 == 0: — True  
4) res = '' + '2\*' = '2\*'  
5) n = 36 // 2 = 18  
3) while 18 > 2 and 18 % 2 == 0: — True  
4) res = '2\*' + '2\*' = '2\*2\*'  
5) n = 18 // 2 = 9  
3) while 9 > 2 and 9 % 2 == 0: — False  
6) d = 3  
7) while 9 > 3: — True  
8) if 9 % 3 == 0: — True  
9) res = '2\*2\*' + str(3) + '\*' = '2\*2\*3\*'  
10) n = 9 // 3 = 3  
7 )while 3 > 3: — False  
13) return '2\*2\*3\*' + str(3)  
 return '2\*2\*3\*3'

7. remove\_digit(537, 3)  
1) def remove\_digit(number=537, digit=3)  
2) res = 0  
3) power = 1  
4) while 537 > 0: — True  
5) cur\_digit = 537 % 10 = 7  
6) if 7 != 3: — True  
7) res = 0 + 7 \* 1 = 7  
8) power = 1 \* 10 = 10  
9) number = 537 // 10 = 53  
4) while 53 > 0: — True  
5) cur\_digit = 53 % 10 = 3  
6) if 3 != 3: — False  
9) number = 53 // 10 = 5  
4) while 5 > 0: — True  
5) cur\_digit = 5 % 10 = 5  
6) if 5 != 3: — True  
7) res = 7 + 5 \* 10 = 57  
8) power = 10 \* 10 = 100  
9) number = 5 // 10 = 0  
4) while 0 > 0: — False  
10) return 57