

**Supplement to the paper “ALGORITHMS. ANAMORPHOSES. ANOMALIES:
Using Mathematics to Make a Dance Performance” by Eka Zharinova**

The code is implemented by Andrei Zharinov.

N = 13 # We need 12, but put 13 to simplify range() expressions

```
# define values for gamma_i
gs = {i:f"g{i}" for i in range(N)}

# define A_i and define G
G = []
A = {}
for i in range(N):
    G = G + [gs[i]]
    A[i] = G
    print(f"A_{i} = {A[i]}")

# define transformations
def inverse(g):
    return g+"-" if g!=gs[0] else g

def phi_1(G):
    res = [inverse(g) for g in G]
    res.reverse()
    return res

def mirror(g):
    return "-" + g if g!=gs[0] else g

def phi_2(G):
    return [mirror(g) for g in G]

print("")

# check transformation properties
assert(inverse(gs[0]) == gs[0])
assert(mirror(gs[0]) == gs[0])
assert(phi_1(phi_2(G)) == phi_2(phi_1(G)))

# display transformations
print(f"\phi_1(G) = {phi_1(G)}")
print(f"\phi_2(G) = {phi_2(G)}")
print("")
print(f"Eka: G + \phi_1(G) + G")
print(G + phi_1(G) + G)
print("")
print(f"Olga: \phi_1(G) + G + \phi_1(G)")
print(phi_1(G) + G + phi_1(G))
print("")
print(f"Eka: \phi_2(G) + \phi_1(\phi_2(G))")
print(phi_1(G) + phi_1(phi_2(G)))
print("")
print(f"Olga: G + \phi_1(G)")
print(G + phi_1(G))
print("")
print(f"Eka: \phi_1(G)")
print(phi_1(G))
print("")
print(f"Olga: \phi_2(\phi_1(G))")
```

```
print(phi_2(phi_1(G)))
```

```
print("")
```

```
# display sequences B_i  
def inverse_sequence(G):  
    return phi_1(G)
```

```
B = {0: [gs[0]]}  
print(f"B_0 = {B[0]}")  
for i in range(1,N):  
    B[i] = A[i] + inverse_sequence(A[i])  
    print(f"B_{i} = {B[i]}")
```

```

A_0 = ['g0']
A_1 = ['g0', 'g1']
A_2 = ['g0', 'g1', 'g2']
A_3 = ['g0', 'g1', 'g2', 'g3']
A_4 = ['g0', 'g1', 'g2', 'g3', 'g4']
A_5 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5']
A_6 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6']
A_7 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7']
A_8 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8']
A_9 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9']
A_10 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9',
'g10']
A_11 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9',
'g10', 'g11']
A_12 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9',
'g10', 'g11', 'g12']

```

```

\phi_1(G) = ['g12-', 'g11-', 'g10-', 'g9-', 'g8-', 'g7-', 'g6-',
'g5-', 'g4-', 'g3-', 'g2-', 'g1-', 'g0']
\phi_2(G) = ['g0', '-g1', '-g2', '-g3', '-g4', '-g5', '-g6', '-g7',
'-g8', '-g9', '-g10', '-g11', '-g12']

```

```

Eka: G + \phi_1(G) + G
['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9', 'g10',
'g11', 'g12', 'g12-', 'g11-', 'g10-', 'g9-', 'g8-', 'g7-', 'g6-',
'g5-', 'g4-', 'g3-', 'g2-', 'g1-', 'g0', 'g0', 'g1', 'g2', 'g3',
'g4', 'g5', 'g6', 'g7', 'g8', 'g9', 'g10', 'g11', 'g12']

```

```

Olga: \phi_1(G) + G + \phi_1(G)
['g12-', 'g11-', 'g10-', 'g9-', 'g8-', 'g7-', 'g6-', 'g5-', 'g4-',
'g3-', 'g2-', 'g1-', 'g0', 'g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6',
'g7', 'g8', 'g9', 'g10', 'g11', 'g12', 'g12-', 'g11-', 'g10-',
'g9-', 'g8-', 'g7-', 'g6-', 'g5-', 'g4-', 'g3-', 'g2-', 'g1-', 'g0']

```

```

Eka: \phi_2(G) + \phi_1(\phi_2(G))
['g12-', 'g11-', 'g10-', 'g9-', 'g8-', 'g7-', 'g6-', 'g5-', 'g4-',
'g3-', 'g2-', 'g1-', 'g0', '-g12-', '-g11-', '-g10-', '-g9-', '-
g8-', '-g7-', '-g6-', '-g5-', '-g4-', '-g3-', '-g2-', '-g1-', 'g0']

```

```

Olga: G + \phi_1(G)
['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9', 'g10',
'g11', 'g12', 'g12-', 'g11-', 'g10-', 'g9-', 'g8-', 'g7-', 'g6-',
'g5-', 'g4-', 'g3-', 'g2-', 'g1-', 'g0']

```

```

Eka: \phi_1(G)
['g12-', 'g11-', 'g10-', 'g9-', 'g8-', 'g7-', 'g6-', 'g5-', 'g4-',
'g3-', 'g2-', 'g1-', 'g0']

```

```

Olga: \phi_2(\phi_1(G))
['-g12-', '-g11-', '-g10-', '-g9-', '-g8-', '-g7-', '-g6-', '-g5-',
'-g4-', '-g3-', '-g2-', '-g1-', 'g0']

```

```

B_0 = ['g0']
B_1 = ['g0', 'g1', 'g1-', 'g0']

```

```

B_2 = ['g0', 'g1', 'g2', 'g2-', 'g1-', 'g0']
B_3 = ['g0', 'g1', 'g2', 'g3', 'g3-', 'g2-', 'g1-', 'g0']
B_4 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g4-', 'g3-', 'g2-', 'g1-',
'g0']
B_5 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g5-', 'g4-', 'g3-',
'g2-', 'g1-', 'g0']
B_6 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g6-', 'g5-',
'g4-', 'g3-', 'g2-', 'g1-', 'g0']
B_7 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g7-', 'g6-',
'g5-', 'g4-', 'g3-', 'g2-', 'g1-', 'g0']
B_8 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g8-',
'g7-', 'g6-', 'g5-', 'g4-', 'g3-', 'g2-', 'g1-', 'g0']
B_9 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9',
'g9-', 'g8-', 'g7-', 'g6-', 'g5-', 'g4-', 'g3-', 'g2-', 'g1-', 'g0']
B_10 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9',
'g10', 'g10-', 'g9-', 'g8-', 'g7-', 'g6-', 'g5-', 'g4-', 'g3-',
'g2-', 'g1-', 'g0']
B_11 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9',
'g10', 'g11', 'g11-', 'g10-', 'g9-', 'g8-', 'g7-', 'g6-', 'g5-',
'g4-', 'g3-', 'g2-', 'g1-', 'g0']
B_12 = ['g0', 'g1', 'g2', 'g3', 'g4', 'g5', 'g6', 'g7', 'g8', 'g9',
'g10', 'g11', 'g12', 'g12-', 'g11-', 'g10-', 'g9-', 'g8-', 'g7-',
'g6-', 'g5-', 'g4-', 'g3-', 'g2-', 'g1-', 'g0']

```