**Projekt 1**

**Opis Algorytmu**

Funkcja Hanoi służy do przenoszenia odpowiedniej ilości krążków na konkretny palik

Powtórzenie linii

„os << a << " " << b << endl; count++;” dwa razy spowodowane jest parzystą ilością krążków, a zmienna count zlicza ilość ruchów.

Zmienne a,b,c oraz n odpowiadają numerom palików a zmienna n odpowiada ilości krążków.

W mainie, zapis:

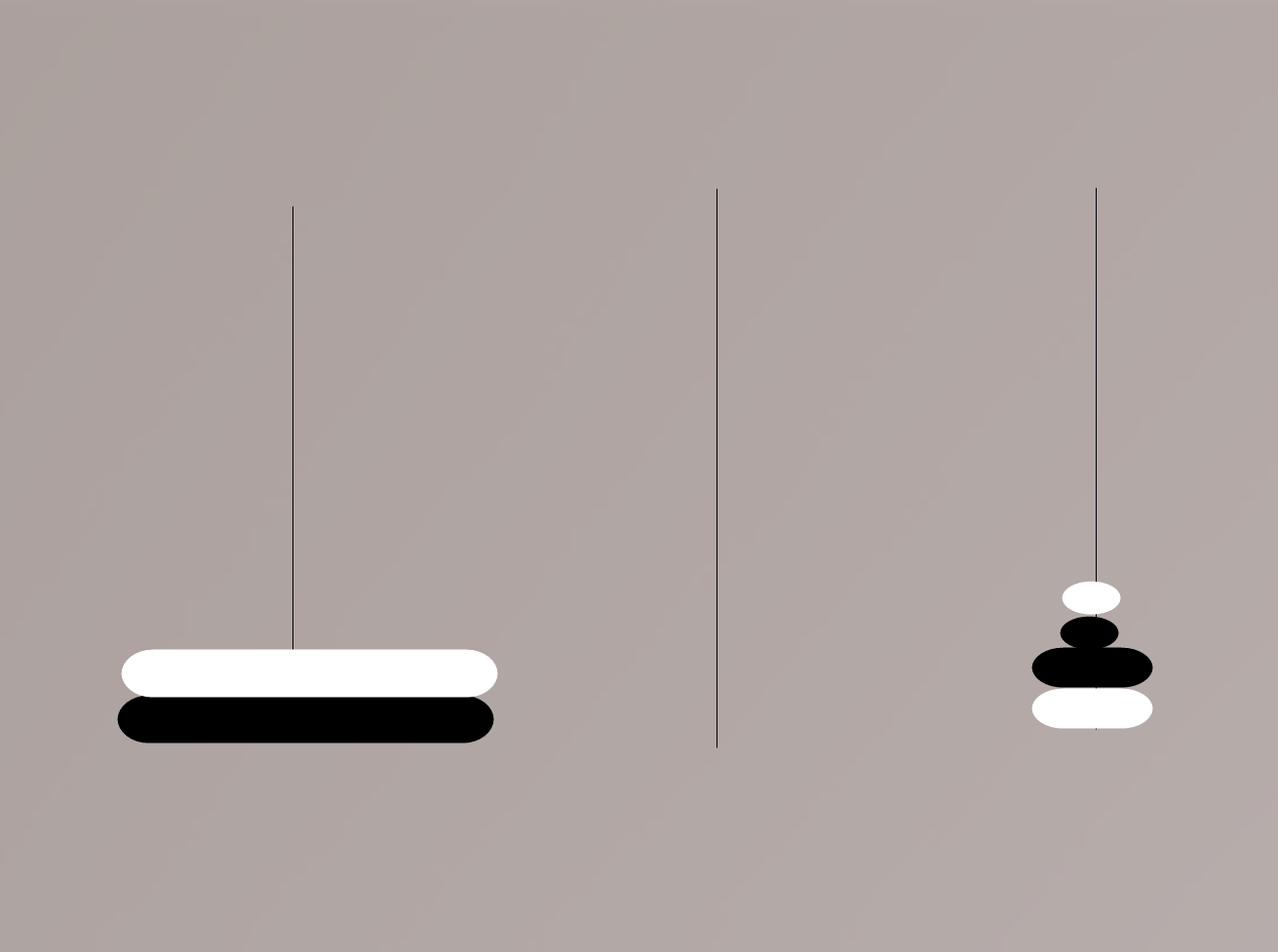
„os<< n\*2 << endl << 3 << endl;”

Powoduje, że zawsze otrzymamy parzystą ilość krążków oraz liczbe palików.

Wszystko zapisywane jest do pliku „wynik.txt”.

Wywołanie funkcji kolor\_hanoi powoduje wywołanie funkcji Hanoi dla n-1 krążków, po wykonaniu funkcji Hanoi mamy następujący obraz:

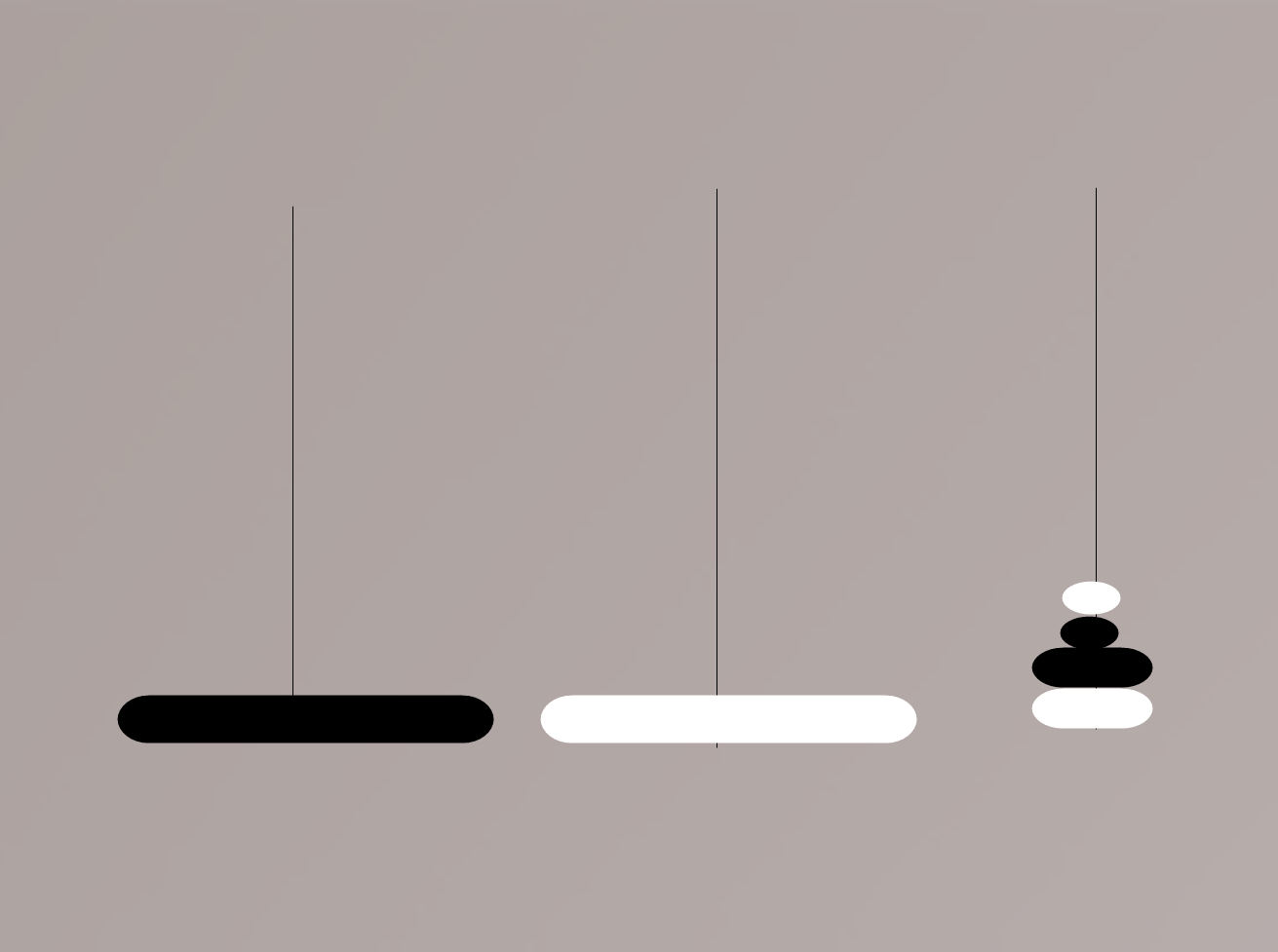
Tutaj dla n=3



Po czym wracamy do funkcji kolor\_hanoi i wykonuje się linijka

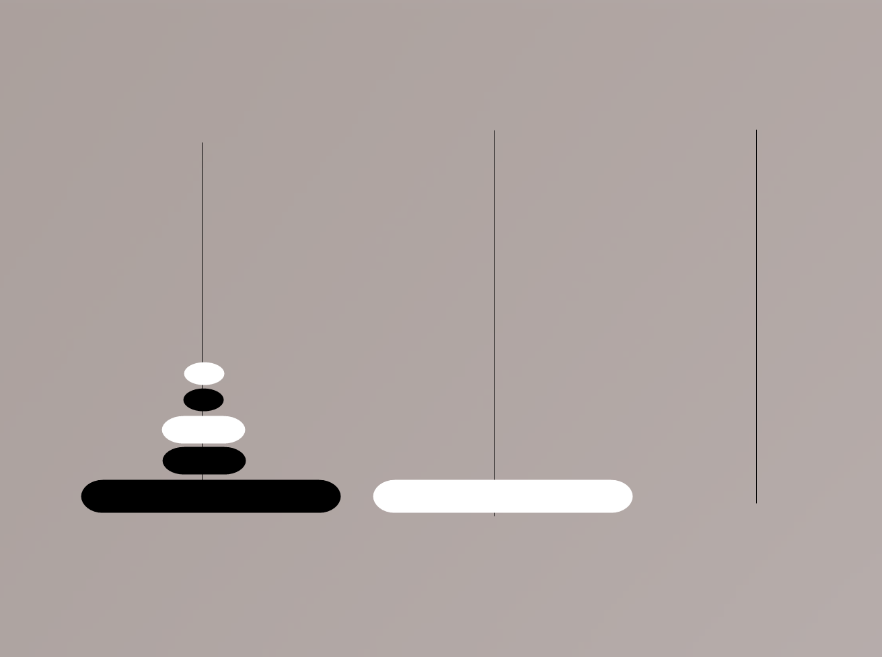
„os << 1 << " " << 2 << endl; count++;”

Powodująca przeniesienie największego krążka na drugi palik:



Teraz wywołujemy funkcję Hanoi

„hanoi(3,1,2,n-1);”, która ustawia wszystkie krążki z palika trzeciego z powrotem na pierwszy.



Dalej wywołujemy funkcję „kolor\_hanoi(1,2,3,n-1);”, która znów prowadzi nas do wywołania hanoi(1,3,2,n-1); (n-1 daje nam w tym przypadku n=1) przez co w funkcji Hanoi od razu przechodzimy do warunku stopu i mamy:

„if(n==1)

{

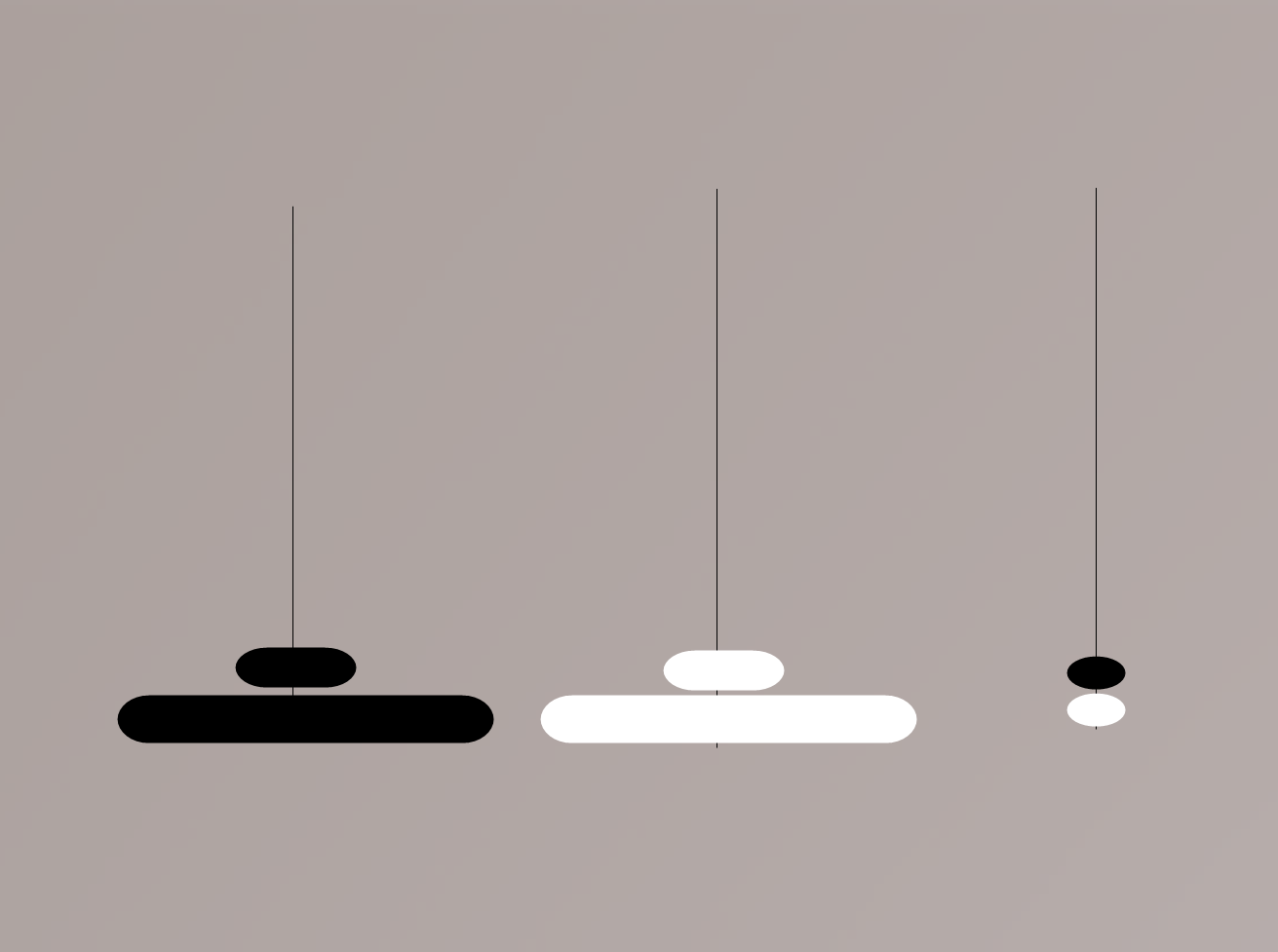
os << a << " " << b << endl; count++;

os << a << " " << b << endl; count++;

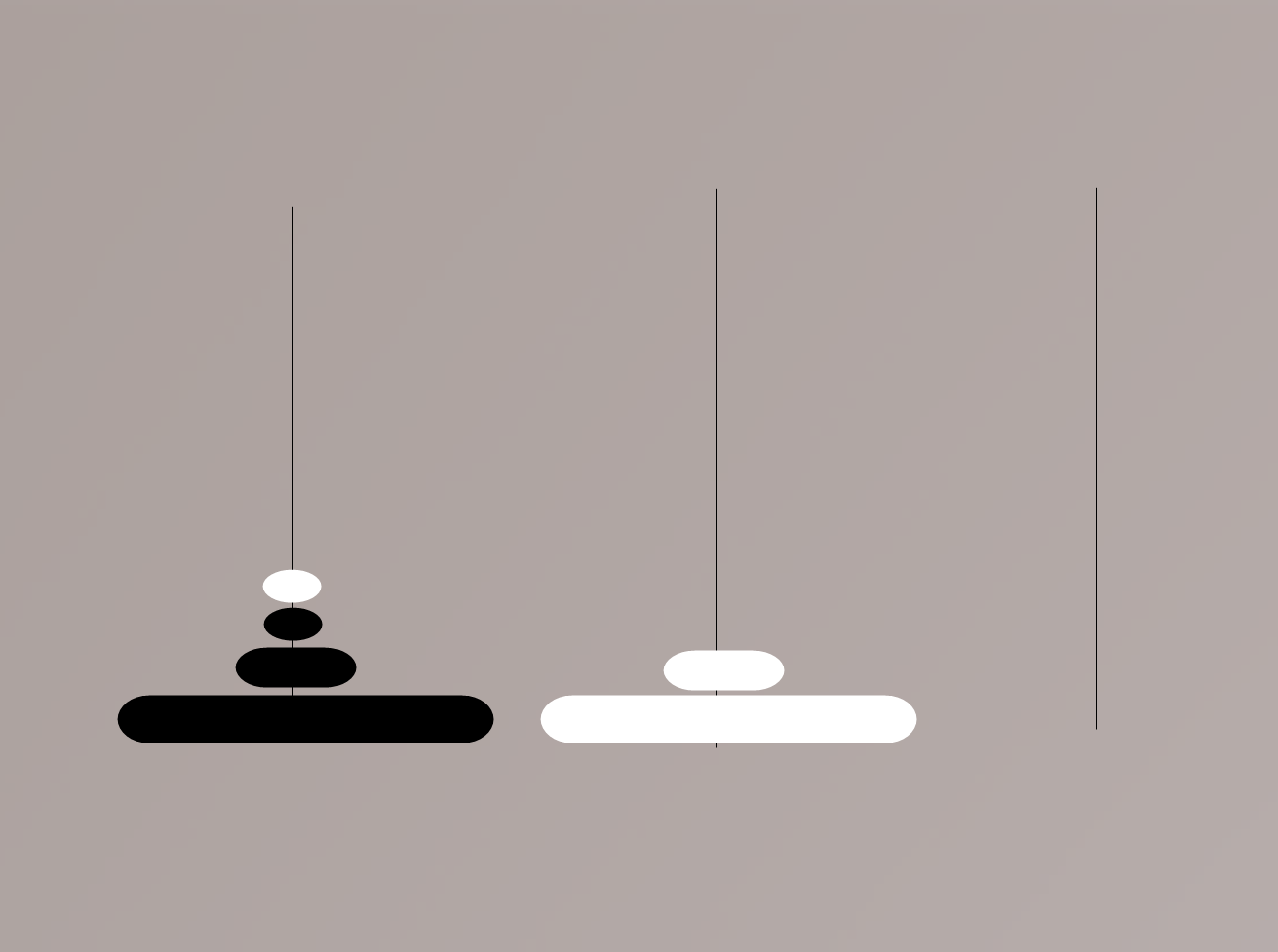
}”

Oraz

„ os << a << " " << b << endl; count++;” – w tym wypadku już w funkcji kolor\_hanoi.



Następnie linijka „hanoi(3,1,2,n-1);” daje taki rezultat:



I wywołuje się funkcja „kolor\_hanoi(1,2,3,n-1);”

(w tym przypadku n będzie równe 1 co spowoduje wykonanie się warunku

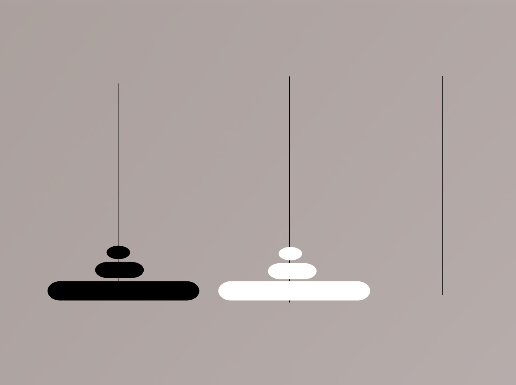
if(n==1)

{

count++;

os << 1 << " " << 2 << endl;

}



Funkcja wykonuje się do momentu, aż zostanie jeden krążek.

Na koniec otrzymujemy na wyjściu standardowym ilość ruchów potrzebną do ułożenia wież oraz kroki, które wklejamy do programu HanoiTwoColor.exe

**Wzory:**

Wzór rekurencyjny dla funkcji Hanoi

F(n) = 2 dla n=1

2\*F(n-1) + 2 dla n > 1

Wzór rekurencyjny dla funkcji kolor\_hanoi

G(n) = 1 dla n =1

G(n-1) + 2\*F(n-1) + 1 dla n > 1

Przykład

Dla n = 5

G(5) = G(4) + 2\*(F(4)) + 1

G(4) = G(3) + 2\*(F(3)) + 1

G(3) = G(2) + 2\*(F(2)) + 1

G(2) = G(1) + 2\*(F(1)) + 1

G(1) = 1 , F(1) = 2 => 1 + 2\*2+1 => 6

G(2) = 6,

G(3) = 6 + 2\*(F(2)) + 1

F(2) = 2\*F(1) + 2 => 2\*2+2 = 6, zatem

G(3) = 6 + 2\*6 + 1 = 19

G(3) = 19,

G(4) = 19 + 2\*(F(3)) + 1

F(3) = 2\*(F(2)) + 2 => 2\*6 + 2 => 14, zatem

G(4) = 19 + 2\*14 + 1 => 48

G(4) = 48

G(5) = 48 + 2\*(F(4)) + 1

F(4) = 2\*(F(3) + 2 => 2\*14 + 2 => 30, zatem

G(5) = 48 + 2\*30 + 1 => 48 + 61 => 109

Wzór ogólny na ilość kroków dla funkcji Hanoi

Wzór ogólny dla funkcji kolor\_hanoi

