

# Zadanie 1

Patryk Lisik

10 Styczeń 2023

## Streszczenie

Mędrzec uczynił sułtanowi znaczną przysługę, za którą sułtan postanowił go wynagrodzić według jego życzenia. Mędrzec odpowiedział: „Nie żądam wiele. Wystarczy, że położysz jedno ziarno pszenicy na pierwszym polu szachownicy, dwa ziarna na drugim polu, dwa razy po dwa ziarna na kolejnym polu i tak dalej, podwajając za każdym razem liczbę ziaren, aż zapelnisz wszystkie pola na szachownicy”. Jak wiele ziaren musi zebrać sułtan, aby zapłacić mędrce za przysługę? Podaj rozwiązanie w zwartej postaci. Jaki typ zmiennej dla tej wartości musi zadeklarować nadworny informatyk sułtana, aby mieć pewność, że poprawnie napisał w języku C++ program obliczający liczbę ziaren pszenicy dla mędrca?

## 1 Obliczenie ilości ziaren

Rozpatrzmy standardową szachownicę  $8 \times 8$ , co oznacza 64 pola. Ilość ziaren na każdym polu można wyrazić jako potęgę liczby 2. Na pierwszym polu znajduje się jedno ziarno ( $2^0$ ), na drugim polu są dwa ziarna ( $2^1$ ), na  $n$ -tym polu znajduje się  $2^{n-1}$  ziarenek. Zakładając przybliżenie  $2^{10} = 10^3$ , ich sumę można wyrazić jako.

$$n^0 + n^1 + \dots + n^{63} = \sum_{n=0}^{63} 2^n = 2^{64} = (2^{10})^6 \cdot 2^4 \approx 16 \cdot 10^{18}$$

## 2 Typ zmiennej w C++

Nawet bez sumowania łatwo zauważyć analogię do liczb binarnych. Każde pole szachownicy oznacza jeden bit w zmiennej bez znaku. Długość zmiennej w języku c++ jest zależna od platformy. Na platformie LP64 wystarczy zmienna typu `unsigned long int`. Na wszystkich platformach można wykorzystać typ `unsigned long long int`. Krótszą bardziej czytelną formę, niezależną od platformy można znaleźć w nagłówku `<stdint.h>` jako `uint64_t`. Wykonanie poniższego kodu można znaleźć tutaj <https://ideone.com/tMhW9w>. Test wykonano na kompilatorze gcc 8.3 na środowisku 64-bit.

```
1 #include <stdint.h>
2 #include <stdint>
3 #include <iostream>
4
5 int main() {
6     long unsigned int seed_count = 0;
7     long unsigned int current_count = 1;
8     for(int i=0; i<64; i++){
```

```
9     seed_count+=current_count;  
10    current_count*=2;  
11 }  
12 std::cout<<"Seeds count: "<<seed_count<<"\n";  
13 return 0;  
14 }
```

Listing 1: Kod obliczający ilość ziarenek.