Emilia Mioduszewska  
Piotr Węglewski

**Task 2**

Algorithm design strategies.

1. Experimental Procedure

We decided to implement a Problem 2 and 3. Profiling is based on time of function execution with 1ms or 1ns accuracy depednig on the settings.

1. Problem 2

Naive approach is based on iterative searching.

Faster solution is based on divide and conquer strategy and it's very similar to binary search.

Results

The result is case of “binary search” was as expected: O(n) complexity for naive approach and O(logn) for divide and conquer strategy. In worth to point out that in case of faster solution the time behaves like a constant value even if the size of an array was increased 4-times.

1. Problem 3

Naive approach is again based on iterative solution. Algorithm iterates through the array and first checks what player does not play current day and then it searches for the opponent.

Divide and conquer approach is implemented nie wiem jak ale daje dobre rezultaty

Results

In case of “tournament problem” we managed to decrease the algorithm complexity from O(n4) to O(n2). Again the divide the conquer strategy introduced a great improvement in time needed to accomplish the task. From the graphs we can see that faster solution for n=4096 is still faster that naive approach for n=1024. And comparing the time for n=1024 the second solution is over 7000 times faster.