

Pomiar czasu za pomocą biblioteki chrono:

<https://en.cppreference.com/w/cpp/chrono>

Biblioteka ta umożliwia wiele operacji na czasie. My wykorzystamy ją tylko do pomiaru czasu trwania operacji.

KOD3:

```
1  #include <chrono>
2  #include <cstdio>
3  #include <windows.h>
4
5  int main(){
6      auto start = std::chrono::steady_clock::now();
7      //długie operacje
8      Sleep(2000);
9      auto end = std::chrono::steady_clock::now();
10
11     printf("Czas trwania: %llu\n", std::chrono::duration_cast<std::chrono::milliseconds>(end - start).count());
12
13     return 0;
14 }
```

Zadania do KOD3:

1. Sprawdź ile trwają na Twoim komputerze operacje: otwarcia i zamknięcia pliku (<https://www.cplusplus.com/reference/fstream/fstream/fstream>)
2. Sprawdź ile trwa na Twoim komputerze wygenerowanie jakiejś większej (np. 40) ilości elementów ciągu fibonacciego.

Wielowątkowe dodawanie dwóch tablic do siebie indeks po indeksie:

KOD4:

```
1  #include <cstdio>
2  #include <cstdlib>
3  #include <time.h>
4  #include <thread>
5
6  #define SIZE 40
7
8  void add(int id, int* a, int* b, int* c){
9      c[id] = a[id] + b[id];
10 }
11
```

```

12 int main(){
13     srand(time(NULL));
14     int a[SIZE];
15     int b[SIZE];
16     int c[SIZE];
17
18     for(int i = 0; i < SIZE; i++){
19         a[i] = rand() % 100 + 1; //1 do 100
20         b[i] = rand() % 100 + 1;
21     }
22
23     //wypisanie na ekranie A
24     for(int i = 0; i < SIZE; i++){
25         printf("%u ", a[i]);
26     }
27     printf("\n");
28
29     //wypisanie na ekranie C
30     for(int i = 0; i < SIZE; i++){
31         printf("%u ", b[i]);
32     }
33     printf("\n");
34
35     std::thread** threads = new std::thread*[SIZE];
36     for(int i = 0; i < SIZE; i++){
37         threads[i] = new std::thread(add, i, a, b, c); //wykorzystuje i jako id danego wątku
38     }
39
40     for(int i = 0; i < SIZE; i++){
41         threads[i]->join();
42     }
43
44     for(int i = 0; i < SIZE; i++){
45         delete threads[i];
46     }
47     delete[] threads;
48
49     //wypisanie na ekranie C
50     for(int i = 0; i < SIZE; i++){
51         printf("%u ", c[i]);
52     }
53
54     return 0;
55 }

```

Powyższy program tworzy tablice o zadanym rozmiarze. Wypełnia losowymi danymi z zakresu 1 do 100. Otwiera tyle wątków ile jest elementów w tablicy. Wątki wykonują dodawanie każdy pod swoim indeksem.

Zadania do KOD4:

1. Dodaj pomiar czasu do powyższego kodu (tak by pomiar obejmował tylko faktyczne dodawanie)
2. Zmodyfikuj program tak by każdy wątek dodawał 10 komórek.
Wątek o id 0 dodaje indeksy 0 z 0, 1 z 1, ... 9 z 9
Wątek o id 1 dodaj indeksy 10 z 10, ... 19 z 19
ltd.
Dostosuj SIZE i ilość wątków tak by to miało sens.
Dodaj pomiar czasu.
3. Odpowiedz na pytanie co trzeba zmienić i dlaczego by rozmiar a, b, c można było pobierać od użytkownika.