

Operacje atomowe realizowane za pomocą `std::atomic`.

Szablon `std::atomic` pozwala wykonywać pewne operacje, których lista znajduje się w dokumentacji (link niżej) w sposób atomowy. Operacje te dotyczą podstawowych typów danych, których lista również znajduje się w linku niżej.

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

Przykład zastosowania operacji atomowych, który sumuje elementy w wektorze do wspólnej zmiennej dla wszystkich wątków:

KOD10:

```
1  #include <cstdio>
2  #include <thread>
3  #include <atomic>
4
5  std::atomic<unsigned long long> sum2(0);
6
7  void sum(unsigned char* data, int id, int count){
8      for(unsigned i = id*count; i<(id+1)*count; i++){
9          sum2+=data[i];
10     }
11 }
12
13 int main(){
14     unsigned char* data = new unsigned char[10000];
15
16     for(unsigned int i = 0; i<10000; i++){
17         data[i] = i;
18     }
19
20     //suma na jednym wqtku
21     unsigned long long sum1 = 0;
22     for(unsigned i = 0; i<10000; i++){
23         sum1+=data[i];
24     }
25     printf("Suma na 1 watku: %llu\r\n", sum1);
26
27     //suma na wielu wqtkach
28     std::thread t1(sum, data, 0, 2000);
29     std::thread t2(sum, data, 1, 2000);
30     std::thread t3(sum, data, 2, 2000);
31     std::thread t4(sum, data, 3, 2000);
32     std::thread t5(sum, data, 4, 2000);
33
34     t1.join();
35     t2.join();
36     t3.join();
37     t4.join();
38     t5.join();
39
40     printf("Suma na 5 watkach: %llu\r\n", sum2.load());
41 }
```

Zadania do KOD10:

1. Opisz czym są operacje atomowe.
2. Dlaczego są tak przydatne w programowaniu wielowątkowym?
3. Przetestuj KOD7/8 zastępując mutex operacjami atomowymi.