Оптимизация хеш-таблицы
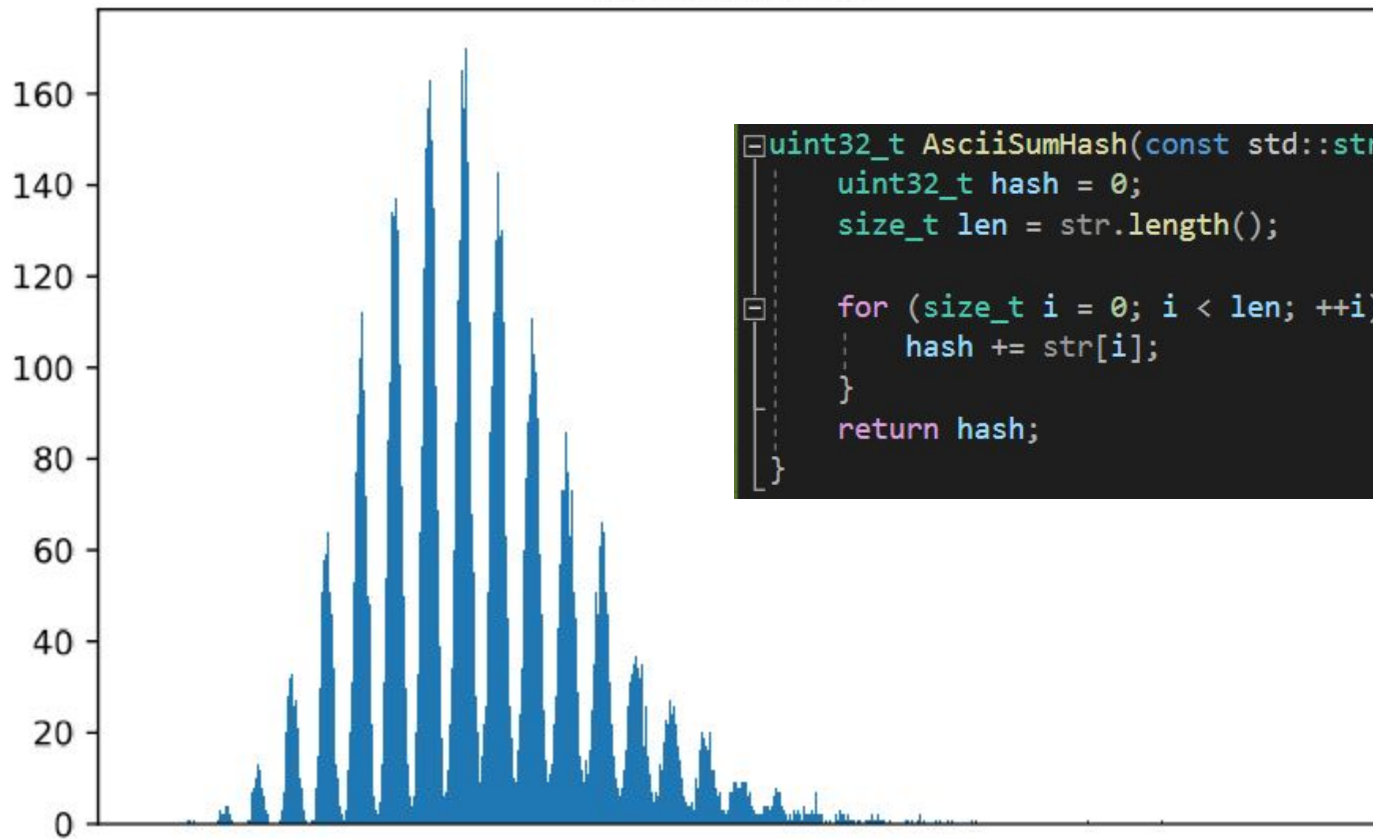
Демина Елизавета, 2020

Hash1

```cpp
uint32_t Hash1(const std::string& str) {
    return 1;
}
```
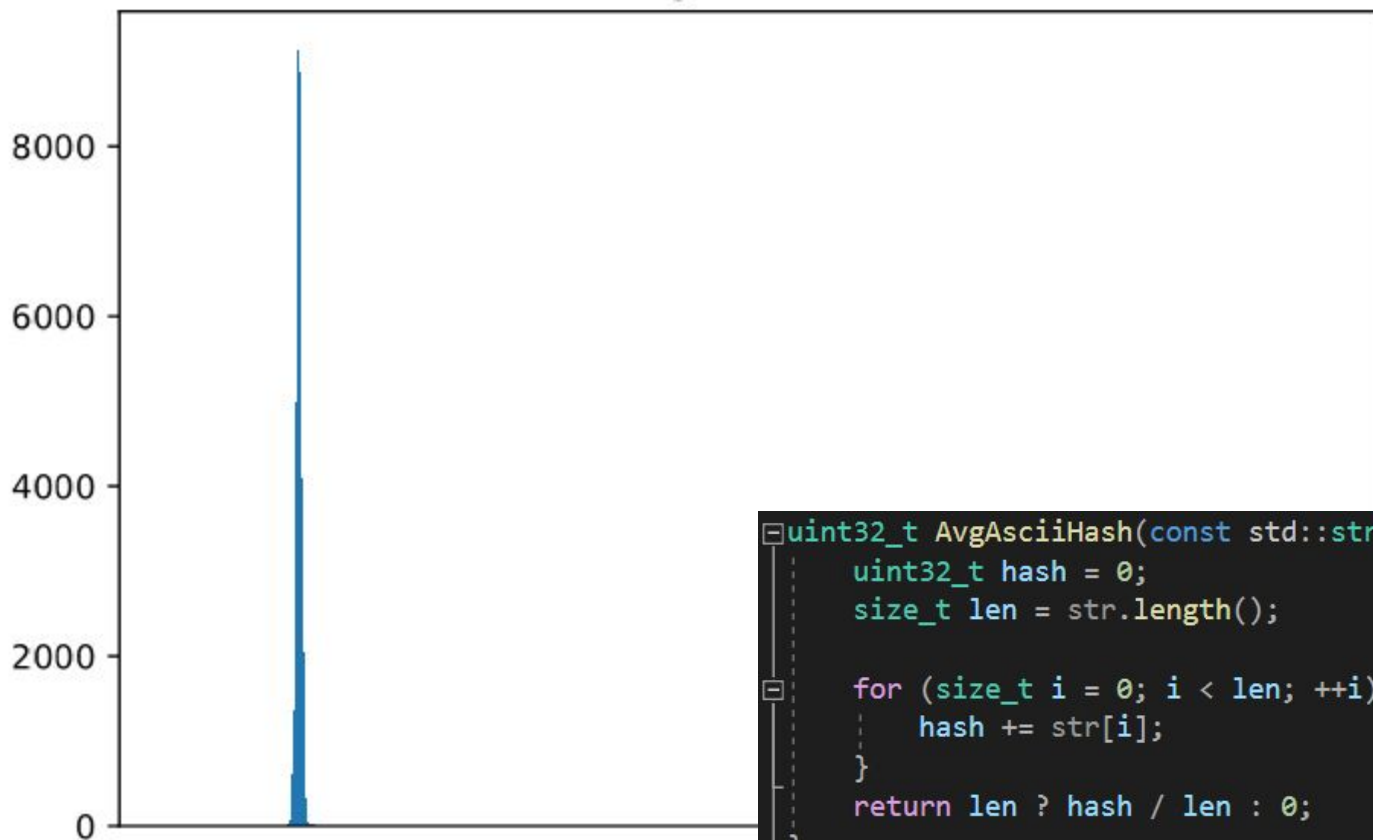
LengthHash


```cpp
uint32_t LengthHash(const std::string& str) {
    return str.length();
}
```
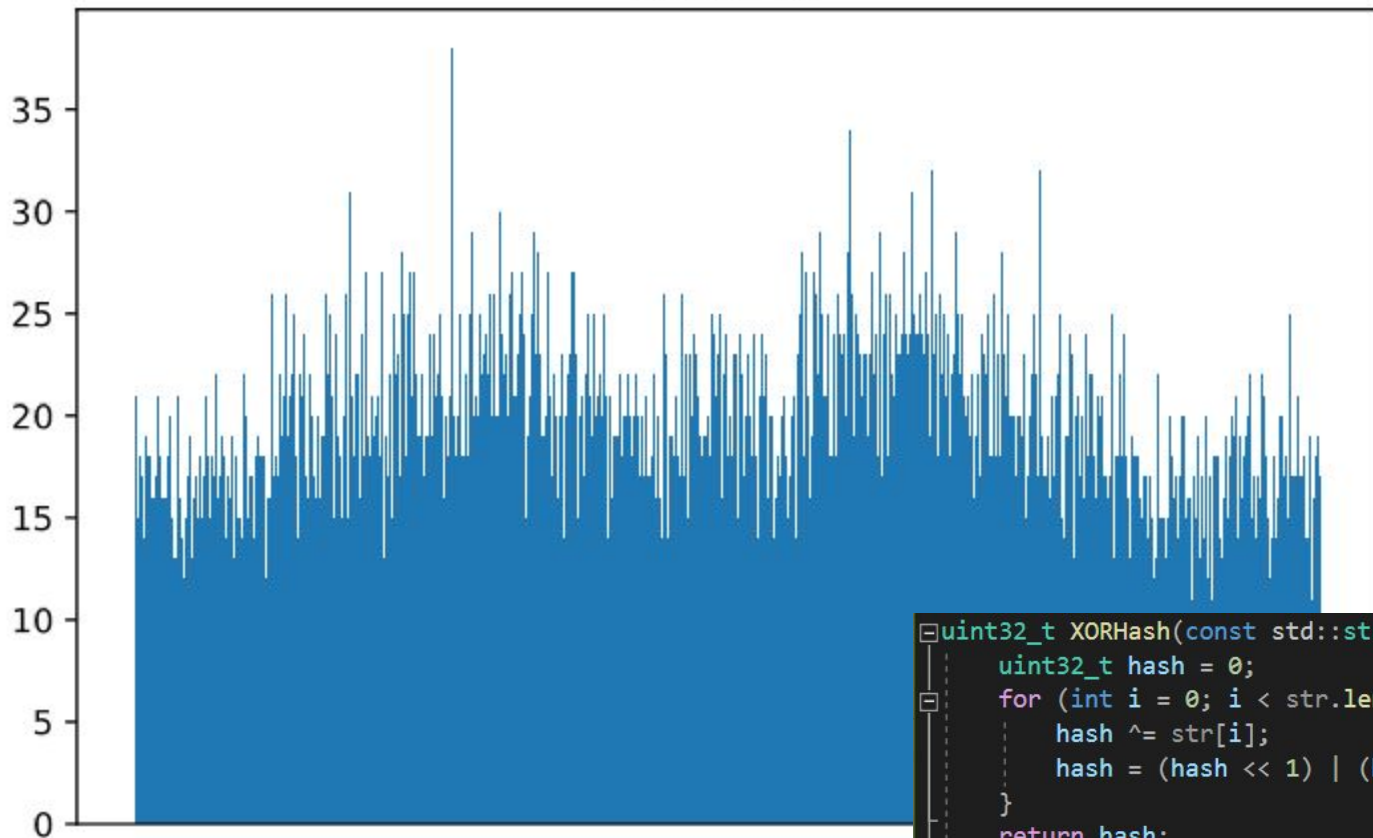
AsciiSumHash


```cpp
uint32_t AsciiSumHash(const std::string& str) {
    uint32_t hash = 0;
    size_t len = str.length();

    for (size_t i = 0; i < len; ++i) {
        hash += str[i];
    }
    return hash;
}
```

AvgAsciiHash

```cpp
uint32_t AvgAsciiHash(const std::string& str) {
    uint32_t hash = 0;
    size_t len = str.length();

    for (size_t i = 0; i < len; ++i) {
        hash += str[i];
    }
    return len ? hash / len : 0;
}
```

XORHash

```cpp
uint32_t XORHash(const std::string& str) {
    uint32_t hash = 0;
    for (int i = 0; i < str.length(); ++i) {
        hash ^= str[i];
        hash = (hash << 1) | (hash >> 31);
    }
    return hash;
}
```

MurMurHash
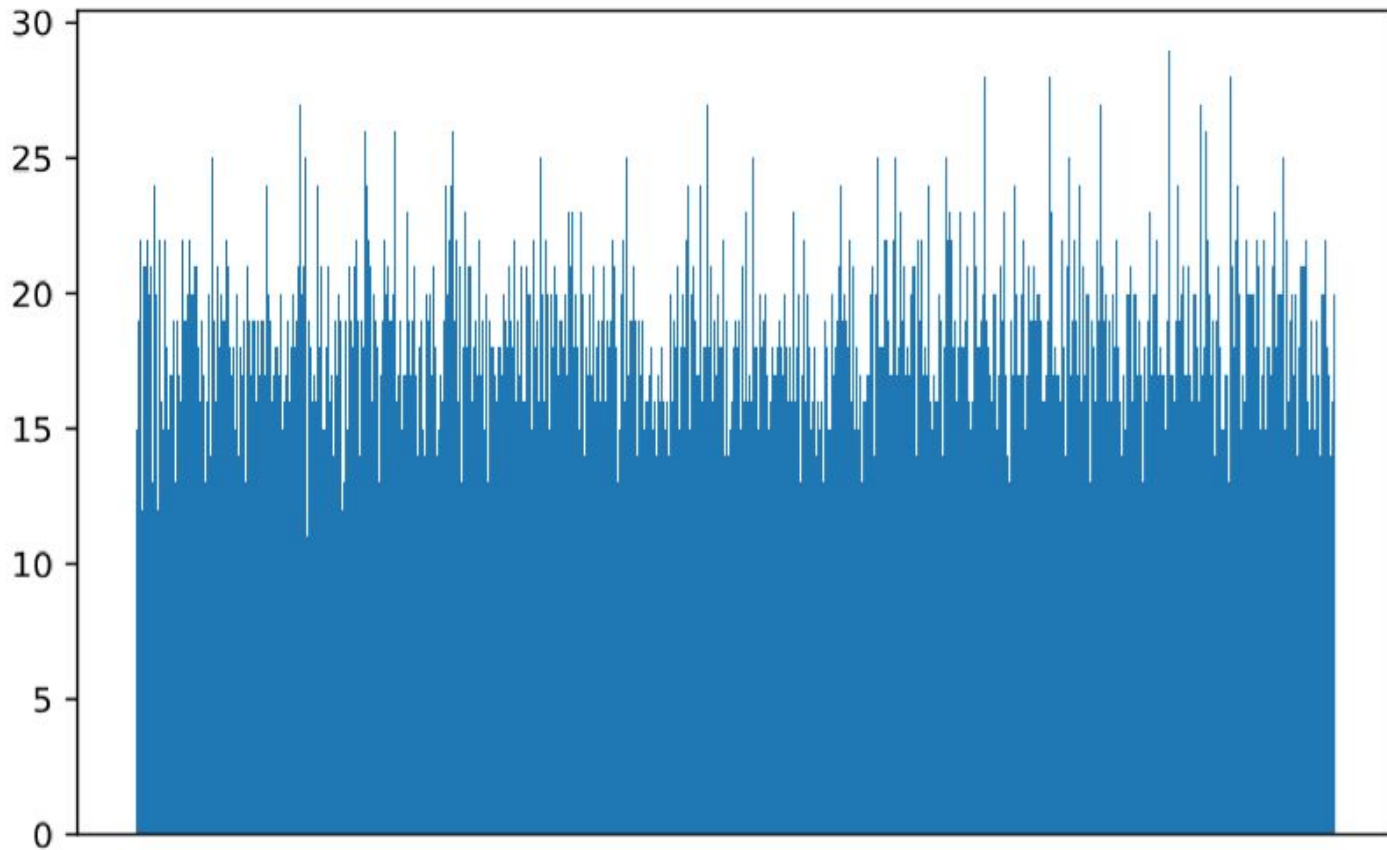
Профилирование хеш-таблицы

std::_Default_allocator_traits<std::allocator<std::_...	3160 (5,66%)	154 (0,28%)	hash_table1.exe
std::_Default_allocator_traits<std::allocator<std::_...	3040 (5,45%)	154 (0,28%)	hash_table1.exe
std::_List_node_emplace_op2<std::allocator<std::_...	2759 (4,94%)	708 (1,27%)	hash_table1.exe
std::basic_string<char,std::char_traits<char>,std::a...	2576 (4,61%)	248 (0,44%)	hash_table1.exe
std::basic_string<char,std::char_traits<char>,std::a...	2363 (4,23%)	485 (0,87%)	hash_table1.exe
XORHash	2292 (4,11%)	175 (0,31%)	hash_table1.exe
AvgAsciiHash	2269 (4,06%)	131 (0,23%)	hash_table1.exe
AsciiSumHash	2160 (3,87%)	133 (0,24%)	hash_table1.exe
std::_String_val<std::_Simple_types<char> >::_Lar...	2056 (3,68%)	1865 (3,34%)	hash_table1.exe
std::_Construct_in_place<std::_List_node<std::basi...	1913 (3,43%)	498 (0,89%)	hash_table1.exe
std::basic_string<char,std::char_traits<char>,std::a...	1734 (3,11%)	358 (0,64%)	hash_table1.exe
std::_Unfancy<char>	1688 (3,02%)	1326 (2,38%)	hash_table1.exe
std::_Container_base12::_Orphan_all	1579 (2,83%)	1024 (1,83%)	hash_table1.exe

Реализации XORHash

```cpp
uint32_t XORHash(const string& str) {
    uint32_t hash = 0;

    for (int i = 0; i < str.length(); ++i) {
        hash ^= str[i];
        hash = (hash << 1) | (hash >> 31);
    }
    return hash;
}
```

```asm
_XORHashAsm proc string:ptr byte, len:dword
    mov ecx, len
    mov edx, dword ptr string

    xor eax, eax            ; eax = 0
    xor ebx, ebx            ; ebx = 0
    hash_loop:
        mov bl, [edx]       ; bl = str[i]
        xor eax, ebx        ; eax ^= ebx
        rol eax, 1          ; eax circular shift right 1
        inc edx             ; edx++
        loop hash_loop

    ret
_XORHashAsm endp
```

C++   Asm

Было

mainCRTStartup	10746 (99,50%)	0 (0,00%)
invoke_main	10745 (99,49%)	0 (0,00%)
main	10736 (99,41%)	18 (0,17%)
HashTable::~HashTable	4421 (40,94%)	0 (0,00%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::`scalar deleting...	4421 (40,94%)	0 (0,00%)
std::_Default_allocator_traits<std::allocator<List<std::basic_string<char,std::char_traits<char...	4421 (40,94%)	0 (0,00%)
std::_Destroy_range<std::allocator<List<std::basic_string<char,std::char_traits<char>,std::all...	4421 (40,94%)	0 (0,00%)
std::vector<List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >,std::a...	4421 (40,94%)	0 (0,00%)
std::vector<List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >,std::a...	4421 (40,94%)	0 (0,00%)
std::vector<List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >,std::a...	4421 (40,94%)	0 (0,00%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::~List<std::basic_...	4420 (40,93%)	4 (0,04%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::RemoveHead	4413 (40,86%)	612 (5,67%)
HashTable::Insert	4245 (39,31%)	15 (0,14%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::Node::`scalar del...	3795 (35,14%)	13 (0,12%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::Insert	2921 (27,05%)	102 (0,94%)
operator delete	2739 (25,36%)	10 (0,09%)
operator delete	2724 (25,22%)	6 (0,06%)
HashTable::GetIndex	2534 (23,46%)	115 (1,06%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::Node::~Node	2528 (23,41%)	8 (0,07%)
std::basic_string<char,std::char_traits<char>,std::allocator<char> >::~basic_string<char,std:...	2513 (23,27%)	32 (0,30%)
XORHash	2324 (21,52%)	679 (6,29%)
operator new	1881 (17,42%)	17 (0,16%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::Node::Node	1844 (17,07%)	20 (0,19%)
std::basic_string<char,std::char_traits<char>,std::allocator<char> >::basic_string<char,std::c...	1779 (16,47%)	57 (0,53%)
HashTable::Contains	1769 (16,38%)	19 (0,18%)
std::_Delete_plain_internal<std::allocator<std::_Container_proxy> >	1551 (14,36%)	17 (0,16%)
std::_Deallocate_plain<std::allocator<std::_Container_proxy> >	1526 (14,13%)	20 (0,19%)

Стало

operator delete	2648 (30,90%)	9 (0,11%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::Node::~Node	2464 (28,75%)	11 (0,13%)
std::basic_string<char,std::char_traits<char>,std::allocator<char> >::~basic_string<char,std::...	2452 (28,61%)	20 (0,23%)
operator new	1864 (21,75%)	15 (0,18%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::Node::Node	1809 (21,11%)	20 (0,23%)
std::basic_string<char,std::char_traits<char>,std::allocator<char> >::basic_string<char,std::c...	1771 (20,67%)	70 (0,82%)
std::_Delete_plain_internal<std::allocator<std::_Container_proxy> >	1478 (17,25%)	18 (0,21%)
std::_Deallocate<8,0>	1465 (17,10%)	13 (0,15%)
std::_Deallocate_plain<std::allocator<std::_Container_proxy> >	1456 (16,99%)	9 (0,11%)
std::_Default_allocator_traits<std::allocator<std::_Container_proxy> >::deallocate	1444 (16,85%)	12 (0,14%)
std::_Container_proxy_ptr12<std::allocator<std::_Container_proxy> >::_Container_proxy_ptr...	1180 (13,77%)	39 (0,46%)
std::_Allocate<8,std::_Default_allocate_traits,0>	971 (11,33%)	14 (0,16%)
std::allocator<std::_Container_proxy>::allocate	968 (11,30%)	13 (0,15%)
std::_Default_allocate_traits::_Allocate	944 (11,02%)	9 (0,11%)
std::basic_string<char,std::char_traits<char>,std::allocator<char> >::_Tidy_deallocate	844 (9,85%)	27 (0,32%)
std::_Container_base12::_Orphan_all	744 (8,68%)	596 (6,96%)
HashTable::Contains	731 (8,53%)	8 (0,09%)
__CheckForDebuggerJustMyCode	570 (6,65%)	570 (6,65%)
HashTable::GetIndex	536 (6,26%)	109 (1,27%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::Contains	411 (4,80%)	28 (0,33%)
List<std::basic_string<char,std::char_traits<char>,std::allocator<char> > >::Find	380 (4,43%)	21 (0,25%)
XORHashAsm	371 (4,33%)	33 (0,39%)
std::operator==<char,std::char_traits<char>,std::allocator<char> >	342 (3,99%)	10 (0,12%)

Замеры

	Od	O1	O2	Asm
Test 1	5842	4117	2140	3915
Test 2	5768	4071	2159	3926
Test 3	5793	4277	2142	3899
Test 4	5788	4134	2151	3904
Test 5	5864	4052	2134	3924
Среднее	5811	4131	2145	3914
Коэфф.ускорения	1.5	1.1	0.5	1