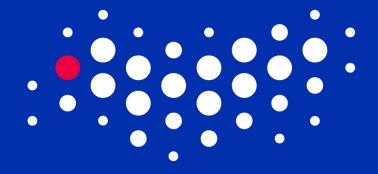


**ITMO UNIVERSITY** 



#### **ITMO UNIVERSITY**

### Ключевое слово auto

Галоев Илькин, гр. v3316



### Auto B C++11

```
В C++11 auto позволяет не указывать тип
переменной явно, говоря компилятору, чтобы
он сам определил фактический тип
переменной, на основе типа
инициализируемого значения.
int main()
   auto i = 10;
   auto d = 3.14;
   auto ld = 3.14L;
   auto S = "c_string";
    return 0;
```





## Auto до C++11

Ключевое слово auto означает, что переменная находится в automatic storage и время жизни такой переменной local lifetime.

int 
$$a = 10$$
;



```
int main(){
    auto i = 10, *p = &i;
    auto x = 2, y = 2.;
}
```

```
std::vector<double> scores;
std::vector<double>::iterator pv = scores.begin();
std::vector<double> scores;
auto pv = scores.begin ();
```

ITSMOre than a UNIVERSITY

```
#include <iostream>
using namespace std;
int main(){
    int x = 100;
    int y = 200;
    int &rx = x;
    auto a = rx;
    a = y;
    cout << x <<
    cout << a <<
```



```
#include <iostream>
using namespace std;
int main(){
    int i = 500;
    const int ci = i, &cr = ci;
    auto b = ci;
    b++;
// ci++;
    auto c = cr;
// cr++;
    auto d = &i;
    cout << *d << '\n';
    auto e = &ci;
    e++;
    (*e)++;
```



```
#include <iostream>
                            #include <iostream>
                            using namespace std;
using namespace std;
                            int main(){
int main(){
                                int i = 300;
    int i = 300;
                                int &p = i;
    int *p = \&i;
                                auto &a = p;
    const auto ptr = p;
                                a = 500;
    ptr++;
                                cout << i << '\n'; //B i 500
                            }
```

#### #include <vector>

```
int main(){
  vector<int> v;

unsigned sz = v.size();
}
```

Member type size\_type is an unsigned integral type

```
auto result = [](const auto& p1, const auto &p2){ return *p1 < *p2; };

template <typename T, typename E>
auto compose(T a, E b) -> decltype(a+b)
{
   return a+b;
}
auto c = compose(2, 3.14);
```



# Спасибо за внимание!