

Handout - Vector, the little container

Declaration & Initialization

A vector<_> contains a variable number of values of the type indicated in <_>.

```
int n;
cin >> n;
vector<int> v(n);
for (int i = 0; i < n; i++) {
    cin >> v[i];
}
```

When we know the values upfront we can directly initialize the vector with values:

```
vector<int> v = {0, 2, 4, 8, 16};
```

Example: Calculate maximum

Assume you are given a A vector<int> v and you have to get the maximum.

```
int max = v[0];
for (int i = 1; i < v.size(); i++) {
    if (v[i] > max)
        max = v[i];
}
cout << "Maximum is: " << max << '\n';
```

We can make the above shorter and nicer by using range based loops.

```
int max = v[0];
// we could also use "auto" instead of int
for (int val : v) {
    if (val > max)
        max = val;
}
cout << "Maximum is: " << max << '\n';
```

Square all elements

Suppose you are given a A vector<int> v and have to square every integer inside it.

```
for (int& val : v) {
    val = val * val;
}
```

Dynamic Vector

We can not only change each element we can also add or remove elements

```
vector<int> v = {0, 2, 4, 8, 16};
v.push_back(32);
// v is now {0, 2, 4, 8, 16, 32}
v.pop_back();
// v is now again {0, 2, 4, 8, 16}
```

Example: Add only integers that are prime

```
int n, tmp;
cin >> n;
```

```
vector<int> v;
for (int i = 0; i < n; i++) {
    cin >> tmp;
    if (is_prime(tmp))
        v.push_back(tmp);
}
```

Example: Voting

Your job is to determine if somebody has more than 50% of all votes. We are given two numbers n, the number of candidates and v, the number of votes. The next v line contains v integers $v_i(0 \leq v_i \leq n)$ the votes cast.

```
int n, v, c;
cin >> n >> v;
// votes[i] is how many votes candidate i has
vector<int> votes(n);
for (int i = 0; i < v; i++) {
    cin >> c;
    votes[c]++;
}
for (auto k : votes) {
    if (k > n/2) {
        cout << "A candidate has won\n";
    }
}
```