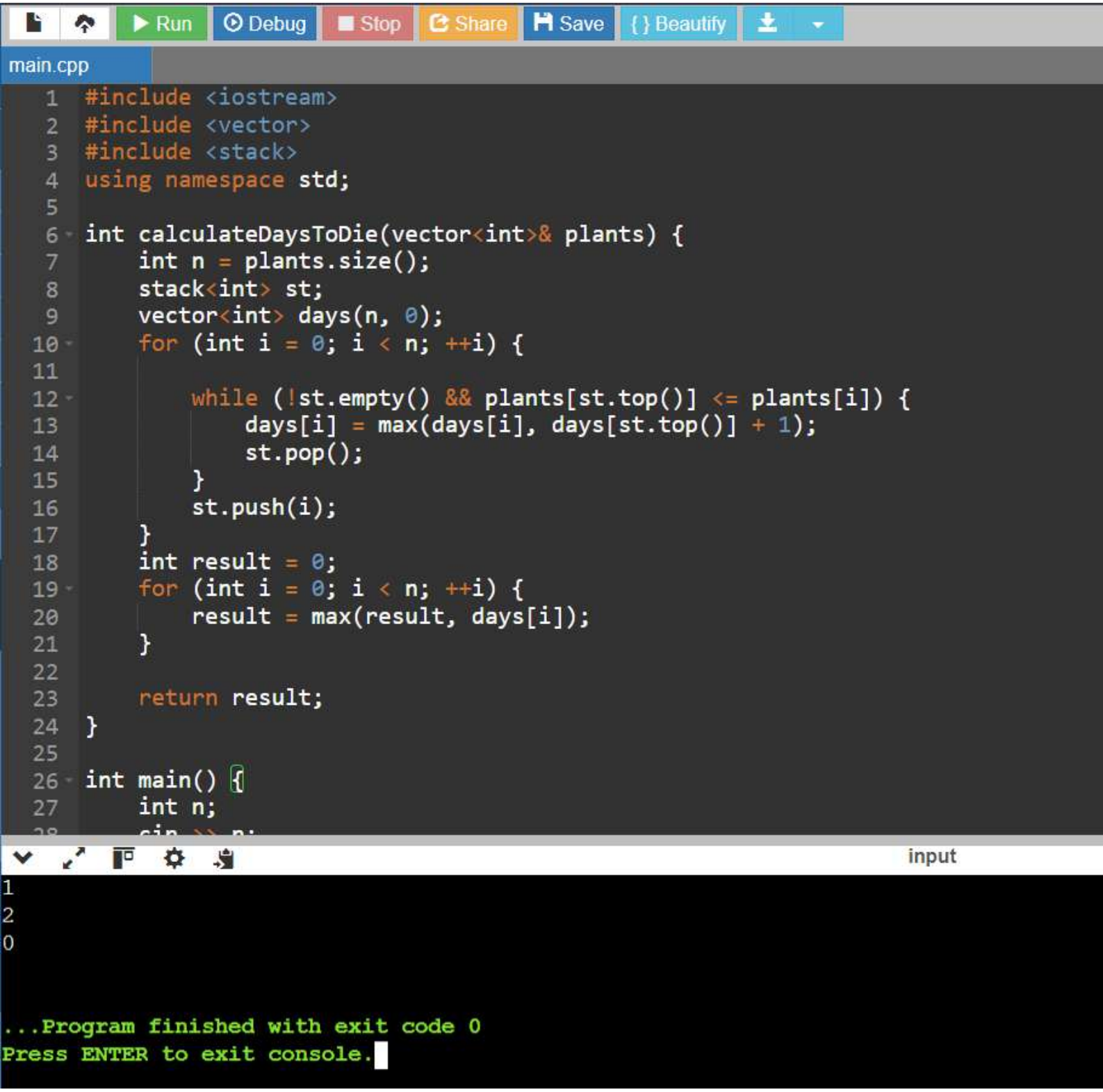
Run
Debug
Stop
Share
Save
{ } Beautify
main.cpp
#include <iostream>
#include <vector>
#include <stack>
using namespace std;
int calculateDaysToDie(vector<int>& plants) {
int n = plants.size();
stack<int> st;
vector<int> days(n, 0);
for (int i = 0; i < n; ++i) {
while (!st.empty() && plants[st.top()] <= plants[i]) {
days[i] = max(days[i], days[st.top()] + 1);
st.pop();
}
st.push(i);
}
int result = 0;
for (int i = 0; i < n; ++i) {
result = max(result, days[i]);
}
return result;
}
int main() {
int n;
input
1
2
0
...Program finished with exit code 0
Press ENTER to exit console.

Run | Debug | Stop | Share | Save | {} Beautify

main.cpp

```cpp
#include <iostream>
#include <vector>
#include <stack>
#include <climits>
using namespace std;

long long calculateSi(int m1, int m2) {
    return (((m1 & m2) ^ (m1 | m2)) * (m1 ^ m2));
}

long long findMaximumSi(const vector<int>& arr) {
    int n = arr.size();
    stack<int> st;
    long long maxSi = LLONG_MIN;

    for (int i = 0; i < n; i++) {
        while (!st.empty() && arr[st.top()] >= arr[i]) {
            int top = st.top();
            st.pop();
            int m1 = arr[top];
            int m2 = (st.empty() ? arr[i] : min(arr[st.top()], arr[i]));
            maxSi = max(maxSi, calculateSi(m1, m2));
        }
        st.push(i);
    }
```

input

```
9
34
2
6
10
12
1024
```

Run | Debug | Stop | Share | Save | { } Beautify

main.cpp

```cpp
#include <stack>
#include <iostream>
using namespace std;

class MinStack {
private:
    stack<int> mainStack;
    stack<int> minStack;

public:
    MinStack() {
    }

    void push(int val) {
        mainStack.push(val);
        if (minStack.empty() || val <= minStack.top()) {
            minStack.push(val);
        }
    }

    void pop() {
        if (!mainStack.empty()) {
            if (mainStack.top() == minStack.top()) {
                minStack.pop();
            }
            mainStack.pop();
        }
    }

```

input

```
-3
0
-2


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify

main.cpp

```cpp
#include <iostream>
#include <vector>
#include <stack>
using namespace std;

vector<int> nextGreaterElements(vector<int>& nums) {
    int n = nums.size();
    vector<int> result(n, -1);
    stack<int> st;
    for (int i = 0; i < 2 * n; i++) {
        while (!st.empty() && nums[st.top()] < nums[i % n]) {
            result[st.top()] = nums[i % n];
            st.pop();
        }
        if (i < n) {
            st.push(i);
        }
    }

    return result;
}

int main() {
    vector<int> nums;
    int n;
    cin >> n;
```

input

```
4
5
6
7
1
6 7 -1 5


...Program finished with exit code 0
Press ENTER to exit console.
```

Run | Debug | Stop | Share | Save | { } Beautify

main.cpp

```cpp
#include <iostream>
#include <stack>
#include <string>
using namespace std;

class TextEditor {
private:
    string S;
    stack<pair<int, string>> history;

public:
    void append(const string& W) {
        history.push({1, W});
        S += W;
    }

    void deleteChars(int k) {
        string deleted = S.substr(S.size() - k, k);
        history.push({2, deleted});
        S.erase(S.size() - k);
    }

    void print(int k) const {
        if (k > 0 && k <= S.size()) {
            cout << S[k - 1] << endl;
        }
    }

    void undo() {
        if (!history.empty()) {
```

input

```
W


...Program finished with exit code 0
Press ENTER to exit console.
```