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

int max(int a, int b, int c) {
    return max(a, max(b, c));
}

int max(int a, int b, int c, int d) {
    return max(a, max(b, c, d));
}
int max(int a, int b, int c, int d, int e) {
    return max(a, max(b, c, d, e));
}
int max(int a, int b, int c, int d, int e, int f) {
    return max(max(a, b, c, d, e), f);
}

int main() {
    cout<<max(1, 2, 3, 4, 5, 6);

    return 0;
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

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

string reverse_str(const string & str)
{
    string ret = str;
    int st = 0, en = (int)str.size()-1;
    while(st < en) {
        char tmp = ret[en];
        ret[en] = ret[st];
        ret[st] = tmp;

        st++, en--;
    }
    return ret;
}
int main() {
    cout<<reverse_str("")<<"\n";
    cout<<reverse_str("abcd")<<"\n";
    cout<<reverse_str("abcde")<<"\n";

    return 0;
```

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

void set_powers(int arr[], int len = 5, int m = 2) {
    arr[0] = 1;

    for (int i = 1; i < len; ++i) {
        arr[i] = arr[i - 1] * m;
    }
}
int main() {
    int arr[100];
    int len, m;
    cin >> len >> m;

    set_powers(arr, len, m);

    for (int i = 0; i < len; ++i)
        cout << arr[i] << " ";
    return 0;
```

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

bool is_prime(int num)
{
    if(num <= 1)
        return false;

    for (int i = 2; i < num; ++i) {
        if(num % i == 0)
            return false;
    }
    return true;
}

int nth_prime(int n)
{
    for (int i = 2; ; ++i) {
        if(is_prime(i)) {
            --n;
            if(n == 0)
                return i;
        }
    }
    return -1;
}

int main() {
    for (int i = 1; i < 20; ++i) {
        cout<<nth_prime(i)<<" ";
```

```
int nth_prime(int n)
{
    for (int i = 2; ; ++i) {
        if(is_prime(i)) {
            --n;
            if(n == 0)
                return i;
        }
    }
    return -1;
}

int main() {
    for (int i = 1; i < 20; ++i) {
        cout<<nth_prime(i)<<" ";
    }

    return 0;
}
```

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

bool starts_with(string input, string pattern, int pos) {
    if (pos + pattern.size() > input.size())
        return false;

    for (int i = 0; i < (int) pattern.size(); ++i) {
        if (pattern[i] != input[i + pos])
            return false;
    }
    return true;
}

string replace_str(string input, string pattern, string to) {
    string res = "";
    for (int pos = 0; pos < (int) input.size(); ++pos) {
        if (starts_with(input, pattern, pos)) {
            res += to;
            pos += (int) pattern.size() - 1;
        } else
            res += input[pos];
    }

    return res;
}
```

```
}

string replace_str(string input, string pattern, string to) {
    string res = "";
    for (int pos = 0; pos < (int) input.size(); ++pos) {
        if (starts_with(input, pattern, pos)) {
            res += to;
            pos += (int) pattern.size() - 1;
        } else
            res += input[pos];
    }

    return res;
}

int main() {
    cout << starts_with("aabcabaaad", "aa", 0) << "\n";
    cout << starts_with("aabcabaaad", "aa", 1) << "\n";
    cout << starts_with("aabcabaaad", "aabcabaaad", 0) << "\n";
    cout << starts_with("aabcabaaad", "baaad", 5) << "\n";
    cout << starts_with("aabcabaaad", "baaad", 4) << "\n";

    cout << replace_str("aabcabaaad", "aa", "x") << "\n";
    cout << replace_str("aabcabaaad", "aa", "aaaa") << "\n";
    cout << replace_str("aabcabaaad", "aa", "") << "\n";

    return 0;
```