

---

# 1

Write the output of the following program. Each line should be written in the corresponding textbox. There are 10 textboxes. If there are more than 10 outputs, only write the first 10. If there are fewer than 10 outputs, write 'x' (lowercase, no quotations, no whitespaces) in the empty textboxes.

Code:

---

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

void func(int &a, int &b, int &c) {
    int temp1 = a;
    int temp2 = b;
    b = c;
    c = temp1;
    a = temp2;
}

int main() {
    int x{4}, y{5}, z{-3};
    int &p = y, *q = &z, *r = &x;
    cout << *r << endl;
    *q+= z - *q;
    cout << ++p << endl;
    cout << (*q)-- << endl;
    func(p, *q, *r);
    cout << p << endl;
    cout << *q << endl;
    cout << *r << endl;
    func(*r, ++p, ++(*q));
    cout << p << endl;
    cout << *r << endl;
    cout << *q << endl;
    cout << *r / *q << endl;
    return 0;
}
```

---

---

Output:

---

4  
6  
-3  
-4  
4  
6  
5  
-3  
6  
0

---

---

## 2

Write the output of the following program. Each line should be written in the corresponding textbox. There are 10 textboxes. If there are more than 10 outputs, only write the first 10. If there are fewer than 10 outputs, write 'x' (lowercase, no quotations, no whitespaces) in the empty textboxes.

Code:

---

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

int main() {
    string str = "Flibbertigibbet";
    for (int i = str.size() - 10; i >= 2; i--) {
        cout << str.substr(i - 2, i + 2) << endl;
    }
    for (int i = 2; i < str.size() - 7; i += 2) {
        cout << str.find(str.at(i), i + 8) << endl;
    }
    for (int i = str.size() - 3; i >= 0; i -= 4) {
        auto pos = find(str.begin() + i, str.end(), 'b');
        size_t index = pos - str.begin();
        cout << str.substr(index - 3, str.size() - i - 2) << endl;
    }
    return 0;
}
```

---

---

Output:

---

bbertig  
ibbert  
libbe  
Flib  
10  
12  
g  
igibb  
libbertig  
Flibbertigibb

---

---

## 3

Write the output of the following program. Each line should be written in the corresponding textbox. There are 10 textboxes. If there are more than 10 outputs, only write the first 10. If there are fewer than 10 outputs, write 'x' (lowercase, no quotations, no whitespaces) in the empty textboxes.

Code:

---

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

class C {
private:
    int z;
public:
    C() : z(5) {}
    C(int val) : z(val) {}
    C(const C &c) : z(c.z) {}
    void H1() { z += 2; }
    void H2() { z -= 1; }
    void H3() { z *= z - 4; }
    void H4() { z /= 3; }
    int H5() const { return z; }
};

int main() {
    C objX;
    C objY(4);
    C objZ(objX);
    objX.H1();
    cout << objX.H5() << endl;
    objX.H2();
    cout << objX.H5() << endl;
    objY.H4();
    cout << objY.H5() << endl;
    objY.H3();
    cout << objY.H5() << endl;
    objZ.H1();
    cout << objZ.H5() << endl;
```

---

```
objZ.H3();  
cout << objZ.H5() << endl;  
C objW(objY);  
objW.H1();  
cout << objW.H5() << endl;  
objW.H4();  
cout << objW.H5() << endl;  
objW.H3();  
cout << objW.H5() << endl;  
objW.H2();  
cout << objW.H5() << endl;  
return 0;
```

```
}
```

---

---

Output:

---

7  
6  
1  
-3  
7  
21  
-1  
0  
0  
-1

---

---

## 4

Write the output of the following program. Each line should be written in the corresponding textbox. There are 10 textboxes. If there are more than 10 outputs, only write the first 10. If there are fewer than 10 outputs, write 'x' (lowercase, no quotations, no whitespaces) in the empty textboxes.

Code:

---

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

void rec(int n) {
    if (n < 0) {
        cout << n + 1 << endl;
    } else {
        if (n % 2 == 0) {
            cout << n << endl;
            n += 3;
        } else {
            cout << n << endl;
            n -= 7;
        }
        rec(n);
        if (n % 2 == 1) {
            cout << n << endl;
        } else if (n % 3 == 2) {
            cout << n << endl;
        } else {
            cout << n << endl;
        }
    }
}

int main() {
    rec(6);
    return 0;
}
```

---



---

Output:

---

6

9

2

5

-1

-2

5

2

9

x

---