

```

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

bool* sieb(unsigned long grenze)
{
    unsigned long i,j;
    bool* feld;

    // neues Speicherfeld erzeugen
    feld = new bool[grenze];

    // gerade Zahlen streichen
    for(i=0; i<grenze; i++)
        if(i<2 || i%2==0)
            *(feld+i) = false;
        else
            *(feld+i) = true;

    // nur ungerade Zahlen können Primzahlen sein
    for(i=3; i < sqrt(double(grenze)); i+=2)
        if(*(feld+i))
            // Vielfache von i löschen
            // bei Quadrat beginnen
            for(j=i*i; j < grenze; j+=i)
                *(feld+j) = false;

    // Zeiger auf Feld zurückgeben
    return feld;
}

void ausgabe(unsigned long grenze, bool *feld)
{
    if(feld == NULL)
        return; // Speicherfehler vermeiden
    for(unsigned long i=0; i<grenze; i++)
        if (*(feld+i) == true)
            cout << i << " ";

    cout << endl;
}

int main()
{
    unsigned long z;
    cout << "Obere Grenze: ";
    cin >> z;
    z++; // da Feldindizierung bei 0 beginnt
    bool* f = sieb(z);
    ausgabe(z,f);
    delete[] f;
}

```