```
p = mn[2];
[2] = mn[mx_pos];
[mx_pos] = tmp;
 (mn[0] > mn[1])

   tmp = mn[0];
   mn[0] = mn[1];
   mn[1] = tmp;



r (int i = 0; i < 3; i++
   cout << mn[i] << " ";

turn 0;
```

```cpp
nclude <iostream>
ing namespace std;

t main()

  const int N = 200;

  int n, a[N];

  cin >> n;
  for (int i = 0; i < n; i++)
      cin >> a[i];


  int mn;
  bool first_time = true;

  for (int i = 0; i < n; i++)
  {
      for (int j = i + 1; j < n; j++)
      {
          int tmp = a[i] + a[j] + j - i;
          if (first_time || tmp < mn)
          {
              mn = tmp;
              first_time = false;
          }
      }
  }
  cout << mn << endl;
  return 0;
```

```
ude <iostream>
 namespace std;

ain()

/ The size of the array should be larger than
/ the maximum value of N
nt arr[1000];

nt N;
in >> N;

or (int i = 0; i < N; i++)
    cin >> arr[i];


or (int i = 0; i < N / 2; i++)

    if (arr[i] != arr[N - 1 - i])
    {
        cout << "NO";
        return 0;
    }

out << "YES";
eturn 0;
```

```
int main()
{
    int n;
    cin >> n;
    const int MAX = 270 + 500 + 1;
    int frequency[MAX] = {0};

    for (int i = 0; i < n; i++)
    {
        int value;
        cin >> value;

        value += 500;
        frequency[value]++;
    }

    int mx_pos = 0;
    for (int i = 1; i < MAX; i++)
    {
        if (frequency[mx_pos] < frequency[i])
            mx_pos = i;
```

```
le <iostream>
amespace std;

n()

 n, x, occurrence[10] = {0};

 >> n;

 (int i = 0; i < n; i++)

  cin >> x;

  while (x)
  {
      int digit = x % 10;
      occurrence[digit]++;
      x /= 10;
  }

 (int i = 0; i <= 9; i++)

  cout << i << " " << occurrence[i] <<
```

```
de <iostream>
namespace std;

in()

nst int N = 201;
 sequence[N], indx;

 occurrence[N * 10] = {0};

 >> indx;
quence[0] = 0;
currence[0] = 1;

 (int i = 1; i <= indx; i++)

  int cur = sequence[i - 1] - (i - 1) - 1;

  if (cur < 0 || occurrence[cur])
      cur = sequence[i - 1] + (i - 1) + 1;

  sequence[i] = cur;
  occurrence[cur] = 1;


ut << sequence[indx];
urn 0;
```

```cpp
int read_array(int arr[])
{
    int len;
    cin >> len;
    for (int i = 0; i < len; ++i)
        cin >> arr[i];

    return len;
}

bool is_palindrome(int arr[], int n)
{
    int st = 0, en = n - 1;
    while (st < en)
    {
        if (arr[st] != arr[en])
            return false;
        st++, en--;
    }
    return true;
}

int main()
{
    int arr[100];
    int len = read_array(arr);
    cout << is_palindrome(arr, len);
    return 0;
}
```

```
nclude <iostream>
ing namespace std;

t menu()

  int choice = -1;
  while (choice == -1)
  {
      if (true)
      {
          cout << "\nMenu;\n";
          cout << "1) Add 2 numbers\n";
          cout << "2) Subtract 2 numbers\n";
          cout << "3) Multiply 2 numbers\n";
          cout << "4) Divide 2 numbers\n";
          cout << "5) Exit\n";
      }

      cout << "\nEnter your menu choice [1 - 5]: ";
      cin >> choice;

      if (!(1 <= choice && choice <= 5))
      {
          cout << "Invalid choice. Try again\n";
          choice = -1;
      }
  }
  return choice;


id read_2_num(double &a, double &b)

  cout << "Enter 2 numbers: ";
  cin >> a >> b;
```

```
void add(double a, double b)
{
    cout << "a + b = " << a + b << "\n";
}

void subtract(double a, double b)
{
    cout << "a - b = " << a - b << "\n";
}

void multiply(double a, double b)
{
    cout << "a * b = " << a * b << "\n";
}

void divide(double a, double b)
{
    if (b != 0)
        cout << "a / b = " << a / b << "\n";
    else
        cout << "can't divide by zero\n";
}

int main()
{
    int total_operations = 0;
    double a, b;

    while (true)
    {
        int choice = menu();

        if (choice == 5)
            break;
```

```
71
72          total_operations++;
73
74          read_2_num(a, b);
75
76          if (choice == 1)
77              add(a, b);
78          else if (choice == 2)
79              subtract(a, b);
80          else if (choice == 3)
81              multiply(a, b);
82          else if (choice == 4)
83              divide(a, b);
84      }
85      cout << "Total operations: " << total_operations;
86
```

```cpp
#include <iostream>
#include <cmath>
using namespace std;
int Max3(int a, int b, int c)
{
    if (a > b && a < c)
    {
        return a;
    }
    if (b > a && b > c)
    {
        return b;
    }

    return c;
}
int Max4(int a, int b, int c, int d)
{
    if (a > b && a < c && a > d)
    {
        return a;
    }
    if (b > a && b > c && a > d)
    {
        return b;
    }
    if (c > a && c > b && c > d)
    {
        return c;
    }
    return d;
}
int Max5(int a, int b, int c, int d, int e)
{
    if (a > b && a < c && a > d && a > e)
```

```
int Max5(int a, int b, int c, int d, int e)
{
    if (a > b && a < c && a > d && a > e)
    {
        return a;
    }
    if (b > a && b > c && a > d && b > e)
    {
        return b;
    }
    if (c > a && c > b && c > d && c > e)
    {
        return c;
    }
    if (d > a && d > b && d > c && d > e)
    {
        return d;
    }
    return e;
}
int Max6(int a, int b, int c, int d, int e, int f)
{
    if (a > b && a < c && a > d && a > e&&a>f)
    {
        return a;
    }
    if (b > a && b > c && a > d && b > e && b > f)
    {
        return b;
    }
    if (c > a && c > b && c > d && c > e && c > f)
    {
        return c;
    }
    if (d > a && d > b && d > c && d > e && d > f)
```

```
x6(int a, int b, int c, int d, int e, int f)
(c > a && c > b && c > d && c > e && c > f)
        ;

(d > a && d > b && d > c && d > e && d > f)

 return d;

(e>a&&e>b&&e>c&&e>d&&e>f)

 return e;


turn f;



n()

 x, y, z, e, h,g;
 >> x >> y >> z >> e >> h>>g;
t <<"========\n";
t << Max3(x, y, z) << "\n";
t <<"========\n";
t << Max4(x, y, z, e) << "\n";
t << "========\n";
t << Max5(x, y, z, e, h) << "\n";
t << "========\n";
t << Max6(x, y, z, e, h,g) << "\n";
turn 0;
```

```
5   {
6       arr[0] = 1;
7
8       for (int i = 1; i < len; ++i)
9       {
10          arr[i] = arr[i - 1] * m;
11      }
12  }
13
14  int main()
15  {
16      int arr[100];
17      int len, m;
18      cin >> len >> m;
19
20      set_powers(arr, len, m);
21
```

```cpp
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 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)
    {
```

```
7           return false;
8
9       for (int i = 2; i < num; ++i)
10      {
11          if (num % i == 0)
12              return false;
13      }
14      return true;
15  }
16
17  int prime(int n)
18  {
19      for (int i = 2;; ++i)
20      {
21          if (is_prime(i))
22          {
23              --n;
24              if (n == 0)
25                  return i;
26          }
27      }
28      return -1;
29  }
30
31  int main()
32  {
33      for (int i = 1; i < 20; ++i)
34      {
35          cout << prime(i) << " ";
```

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

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

int main()
{
    cout <<rev("") << "\n";
    cout << rev("abc") << "\n";
    cout << rev("abcde") << "\n";

    return 0;
}
```

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

int main()
{
    const int N = 500 + 1;
    int n, q, x, ans[N];

    for (int i = 0; i < N; i++)
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cin >> x;
        ans[x] = i;
    }
    int num;
    cin >> q;
    while (q--)
    {
        cin >> num;
        cout << ans[num] << endl;
    }
    return 0;
```

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

int main()
{
    int n, q, a[200];

    cin >> n;
    for (int i = 0; i < n; i++)
        cin >> a[i];

    cin >> q;
    while (q--)
    {
        int num;
        cin >> num;

        int pos = -1;

        for (int i = n - 1; i >= 0; --i)
        {
            if (a[i] == num)
            {
                pos = i;
                break;
            }
        }
        cout << pos << "\n";
    }
    return 0;
}
```

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

int main()
{
    const int N = 500 + 1;
    int n, q, x, ans[N];

    for (int i = 0; i < N; i++)
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cin >> x;
        ans[x] = i;
    }
    int num;
    cin >> q;
    while (q--)
    {
        cin >> num;
        cout << ans[num] << endl;
    }
    return 0;
}
```

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

int main()
{
    const int N = 100;
    int n, a[N], mn = 10000, mx = -1;
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cin >> a[i];
        if (a[i] < mn)
        {
            mn = a[i];
        }
        if (a[i] > mx)
        {
            mx = a[i];
        }
    }

    for (int i = 0; i < n; i++)
    {
        if (a[i] == mn)
            a[i] = mx;
        else if (a[i] == mx)
            a[i] = mn;
    }
    for (int i = 0; i < n; i++)
    {
        if (i)
            cout << " ";
        cout << a[i];
    }
    return 0;
```

```
nt main()

    int n, tmp, mn[3];
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        int value;
        cin >> value;
        if (i < 3)
            mn[i] = value;
        else
        {
            int mx_pos = 0;
            for (int j = 1; j < 3; ++j)
            {
                if (mn[mx_pos] < mn[j])
                    mx_pos = j;
            }

            if (value < mn[mx_pos])
                mn[mx_pos] = value;
        }
    }
    int mx_pos = 0;
    for (int j = 1; j < 3; ++j)
    {
        if (mn[mx_pos] < mn[j])
            mx_pos = j;
    }
```