

Задача 1.

```
#include<iostream>

using namespace std;

int main() {
    int n;
    cin >> n;
    if (n - 10 > n / 2) {
        cout << n - 10;
    }
    else {
        cout << n / 2;
    }
}
```

Задача 2.

```
#include<iostream>

using namespace std;

int main() {
    int h, d, need = 1;
    int k = 0;
    cin >> h >> d;
    while (h - k * d > 0) {
        need = need * (h - k * d);
        k++;
    }
    cout << need;
}
```

Задача 3.

```
#include<iostream>

using namespace std;

int sec[864000];

int main() {

    int k, in, out, last_time = 0, max_count = 0, max_in = 0, max_out = 0;
    cin >> k;
    //Считываем данные и ищем последний уход
    for (int i = 0; i < k; i++) {
        cin >> in >> out;
        if (out > last_time) {
            last_time = out;
        }
        for (int j = in - 1; j < out; j++) {
            sec[j]++;
        }
    }
    //ищем мксимальное кол-во посетителей и время максимального захода
    for (int i = 0; i < last_time; i++) {
        if (sec[i] > max_count) {
```

```

        max_count = sec[i];
        max_in = i;
    }
}
cout << max_count << endl;
// ищем время выхода максимума
for (int i = max_in; i <= last_time; i++) {
    max_out = i;
    if (sec[i] < max_count) {
        break;
    }
}
cout << max_in + 1 << " " << max_out;
}

```

Задача 4.

```

#include<iostream>
#include<string>

```

```

using namespace std;

```

```

int main() {
    string need;

    getline(cin, need);
    if (need.at(0) == '5') {
        cout << "6 6";
        return 0;
    }
    else {
        if (need.at(2) == '3') {
            cout << "11 ж";
            return 0;
        }
    }
    cout << "OK";
}

```

Задача 6.

```

#include<iostream>
#include<string>

```

```

using namespace std;

```

```

int a[9][9];

```

```

int main() {
    int horse_x, horse_y, lad_x, lad_y, queen_x, queen_y;
    int n = 8;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            a[i][j] = 0;
        }
    }

    string need, need2, need3;

```

```
cin >> need;
if (need.at(0) == 'A') {
    queen_x = 0;
}
if (need.at(0) == 'B') {
    queen_x = 1;
}
if (need.at(0) == 'C') {
    queen_x = 2;
}
if (need.at(0) == 'D') {
    queen_x = 3;
}
if (need.at(0) == 'E') {
    queen_x = 4;
}
if (need.at(0) == 'F') {
    queen_x = 5;
}
if (need.at(0) == 'G') {
    queen_x = 6;
}
if (need.at(0) == 'H') {
    queen_x = 7;
}
if (need.at(1) == '1') {
    queen_y = 0;
}
if (need.at(1) == '2') {
    queen_y = 1;
}
if (need.at(1) == '3') {
    queen_y = 2;
}
if (need.at(1) == '4') {
    queen_y = 3;
}
if (need.at(1) == '5') {
    queen_y = 4;
}
if (need.at(1) == '6') {
    queen_y = 5;
}
if (need.at(1) == '7') {
    queen_y = 6;
}
if (need.at(1) == '8') {
    queen_y = 7;
}

cin >> need2;
need = need2;
if (need.at(0) == 'A') {
    lad_x = 0;
}
```

```
if (need.at(0) == 'B') {
    lad_x = 1;
}
if (need.at(0) == 'C') {
    lad_x = 2;
}
if (need.at(0) == 'D') {
    lad_x = 3;
}
if (need.at(0) == 'E') {
    lad_x = 4;
}
if (need.at(0) == 'F') {
    lad_x = 5;
}
if (need.at(0) == 'G') {
    lad_x = 6;
}
if (need.at(0) == 'H') {
    lad_x = 7;
}
if (need.at(1) == '1') {
    lad_y = 0;
}
if (need.at(1) == '2') {
    lad_y = 1;
}
if (need.at(1) == '3') {
    lad_y = 2;
}
if (need.at(1) == '4') {
    lad_y = 3;
}
if (need.at(1) == '5') {
    lad_y = 4;
}
if (need.at(1) == '6') {
    lad_y = 5;
}
if (need.at(1) == '7') {
    lad_y = 6;
}
if (need.at(1) == '8') {
    lad_y = 7;
}
cin >> need3;
need = need3;
if (need.at(0) == 'A') {
    horse_x = 0;
}
if (need.at(0) == 'B') {
    horse_x = 1;
}
if (need.at(0) == 'C') {
    horse_x = 2;
}
}
```

```

if (need.at(0) == 'D') {
    horse_x = 3;
}
if (need.at(0) == 'E') {
    horse_x = 4;
}
if (need.at(0) == 'F') {
    horse_x = 5;
}
if (need.at(0) == 'G') {
    horse_x = 6;
}
if (need.at(0) == 'H') {
    horse_x = 7;
}
if (need.at(1) == '1') {
    horse_y = 0;
}
if (need.at(1) == '2') {
    horse_y = 1;
}
if (need.at(1) == '3') {
    horse_y = 2;
}
if (need.at(1) == '4') {
    horse_y = 3;
}
if (need.at(1) == '5') {
    horse_y = 4;
}
if (need.at(1) == '6') {
    horse_y = 5;
}
if (need.at(1) == '7') {
    horse_y = 6;
}
if (need.at(1) == '8') {
    horse_y = 7;
}

for (int i = 0; i < 8; i++) {
    a[i][lad_x] = 1;
}
for (int i = 0; i < 8; i++) {
    a[lad_y][i] = 1;
}

for (int i = 0; i < 8; i++) {
    a[i][queen_x] = 1;
}
for (int i = 0; i < 8; i++) {
    a[queen_y][i] = 1;
}
int need_x = queen_x, need_y = queen_y;
while (true) {
    a[need_y][need_x] = 1;
}

```

```

        need_y--;
        need_x++;
        if (need_y < 0 || need_x > 8) {
            break;
        }
    }
    need_x = queen_x, need_y = queen_y;
    while (true) {
        a[need_y][need_x] = 1;
        need_y++;
        need_x++;
        if (need_y > 8 || need_x > 8) {
            break;
        }
    }
    need_x = queen_x, need_y = queen_y;
    while (true) {
        a[need_y][need_x] = 1;
        need_y++;
        need_x--;
        if (need_y > 8 || need_x < 0) {
            break;
        }
    }
    while (true) {
        a[need_y][need_x] = 1;
        need_y--;
        need_x--;
        if (need_y < 0 || need_x < 0) {
            break;
        }
    }
}

if (horse_x - 2 > 0 || horse_y + 1 < 8) {
    a[horse_y + 1][horse_x - 2] = 1;
}
if (horse_x - 2 > 0 || horse_y - 1 > 0) {
    a[horse_y - 1][horse_x - 2] = 1;
}
if (horse_x - 1 > 0 || horse_y - 2 > 0) {
    a[horse_y - 2][horse_x - 1] = 1;
}
if (horse_x + 1 < 8 || horse_y - 2 > 0) {
    a[horse_y - 2][horse_x + 1] = 1;
}
if (horse_x + 2 < 8 || horse_y - 1 > 0) {
    a[horse_y - 1][horse_x + 2] = 1;
}
if (horse_x + 2 < 8 || horse_y + 1 < 8) {
    a[horse_y + 1][horse_x + 2] = 1;
}
if (horse_x + 1 < 8 || horse_y + 2 < 8) {
    a[horse_y + 2][horse_x + 1] = 1;
}
if (horse_x - 1 > 0 || horse_y + 2 < 8) {
    a[horse_y + 2][horse_x - 1] = 1;
}

```

```
}
int sum = 0;
a[horse_y][horse_x] = 1;
a[queen_y][queen_x] = 1;
a[lad_x][lad_y] = 1;
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {
        sum += a[i][j];
    }
}

cout << 64 - sum;
}
```