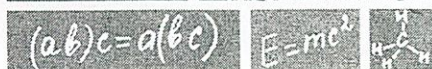




ОТРАСЛЕВАЯ
ОЛИМПИАДА
ШКОЛЬНИКОВ

Использовать только эту сторону листа,
обратная сторона не проверяется!



ШИФР

4618

Класс 11

Вариант

Дата Олимпиады 17.02.2017

Площадка написания КНИТУ

Задача	1	2	3	4	5	6	Σ		Подпись
							Цифрой	Прописью	
Оценка	1	3	5	0	20	0	29	Двадцать девять	

1. $(K \vee L) \wedge (M \vee N) = 1$

Данное выражение может быть равно 1 тогда, когда $\neg(K \vee L) = 1$ или $(M \vee N) = 1$, или и $(K \vee L)$, и $(M \vee N)$ равно 1.

Рассмотрим значение, когда $(K \vee L) = 1$, а $(M \vee N) \neq 1$.

т.е. $K=1$
 $L=1$, а M и N любое, но $(M \vee N) \neq 1$

таких вариантов 3

Также рассмотрим значение, когда $(M \vee N) = 1$, а $(K \vee L) \neq 1$, т.е. $M=1, N=1$, а K и L любое, но $(K \vee L) \neq 1$.

таких вариантов 3.

Кроме того остается вариант, когда и $(M \vee N) = 1$, и $(K \vee L) = 1$. Такой случай единственной и он выполняется при значениях $K=1, L=1, M=1, N=1$.

Отсюда следует, что ответ $\neq$.

Ответ: $\neq$.

2. Переведем все числа последовательности в десятичную систему счисления считая, что они записаны в системе счисления еж с основанием 6.

Получим.

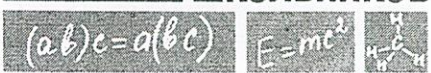
649, 659, 673, 675, 679, 683, 437.

Все числа в данной последовательности являются простыми

Ответ: все числа простые



```
5. #include <iostream>
#include <vector>
#include <algorithm>
#include <cmath>
using namespace std;
int main() {
    int n, ans = 0;
    cin >> n;
    vector<int> h(n, 0);
    for (int i = 0; i < n; i++) {
        vector<pair<int, int>> s;
        for (int j = 0; j < n; j++) {
            int x;
            cin >> x;
            if (x == 0) {
                h[j] = 0;
            } else {
                h[j]++;
            }
        }
        int len = 0;
        while (!s.empty() && s.back().first >= h[j]) {
            len += s.back().second;
            ans = max(ans, (len * s.back().first));
            s.pop_back();
        }
        s.push_back(make_pair(h[j], len));
    }
    int len1 = 0;
    while (!s.empty()) {
        len1 += s.back().second;
        ans = max(ans, (len1 * s.back().first));
        s.pop_back();
    }
    cout << ans;
    return 0;
}
```

3.1 #include

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <cmath>
```

```
using namespace std;
```

```
int n;
long long ans = 0;
vector<vector<int>> g;
```

```
void dfs(int x, int d = 1) {
    for (int i = 0; i < g[x].size(); i++) {
        int to = g[x][i];
        dfs(to, d + 1);
    }
    if (d > n) {
        ans++;
        return;
    }
}
```

```
}
```

```
int main() {
```

```
    cin >> n;
```

```
    g.resize(n);
```

```
    g.resize(10);
```

```
    g.push_back(g);
```

```
    g[1].push_back(1);
```

```
    g[1].push_back(3);
```

```
    g[1].push_back(7);
```

```
    g[1].push_back(9);
```

```
    g[2].push_back(3);
```

```
    g[2].push_back(9);
```

```
    g[3].push_back(1);
```

```
    g[3].push_back(7);
```

```
    g[4].push_back(1);
```

```
    g[4].push_back(3);
```

```
    g[4].push_back(7);
```

```
    g[5].push_back(3);
```

```
    g[5].push_back(9);
```

```
    g[6].push_back(1);
```

```
    g[6].push_back(7);
```

```
    g[7].push_back(1);
```

```
    g[7].push_back(3);
```

```
    g[7].push_back(9);
```

```
    g[8].push_back(7);
```

```
    g[8].push_back(9);
```

```
    g[9].push_back(7);
```

```
    for (int i = 1; i < 10; i++) {
        dfs(i);
    }
```

```
    cout << ans; return 0;
```

```
}
```

3.2.

Ответ: 21

```

6. #include <iostream>
    #include <vector>
    #include <algorithm>
    #include <cmath>

```

```
using namespace std;
```

```

int main() {
    vector<int> a(10);
    for(int i=0; i<10; i++) {
        cin >> a[i];
    }
    int sum;
    cin >> sum;
    vector<vector<int>> dp(11, vector<int> (sum+1, 0)); 501, 0); 5000+1, 0);
    dp[0][0] = 1;
    for(int i=1; i<10; i++) {
        for(int j=0; j<500; j++) {
            if(j < a[i-1]) {
                continue;
            }
            dp[i][j] += dp[i-1][j - a[i-1]];
        }
    }
    for(int j=0; j<500; j++) 5000; j++) {
        dp[0][0] = 1;
        for(int i=1; i<10; i++) {
            dp[i][j] = dp[i-1][j];
            if(j < a[i-1]) {
                continue;
            }
            dp[i][j] += dp[i-1][j - a[i-1]];
        }
    }
    cout << "Контрольная сумма: " << sum << endl;
    if(sum > 5000 || dp[10][sum] == 0) {
        cout << "Ответ не сформирован";
        return 0;
    }
    cout << "Ответ сформирован";
    return 0;
}

```