Олимпиада СПбГУ по информатике 2020/21 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	20	25	445

Task A ()

```
n = int(input())
a = [0, 2, 3, 4, 5, 6, 7, 8, 9]

if n <= 9:
    print(n)
else:
    print(a[(n - 10) % 9])
#for n in arange(1, 1000):
#    s = 0
#    d = 1
#    for i in range(1, n + 1):
#        s += i * d
#        d *= 10
#    print(s // (d // 10) % 10)
```

Task B ()

```python
n, k = map(int, input().split())
line = input()
d = [[0, "", 0] for _ in range(n + 1)]
d[0] = [1, line[0], 1]

for i in range(1, n):
    if d[i - 1][2] < k and (line[i] in d[i - 1][1]):
        d[i] = [d[i - 1][0], d[i-1][1], d[i - 1][2] + 1]
    elif d[i - 1][2] >= k or (not line[i] in d[i-1][1] and len(d[i-1][1]) >= 3):
        d[i] = [d[i - 1][0] + 1, line[i], 1]
    else:
        d[i] = [d[i - 1][0], d[i - 1][1] + line[i], d[i - 1][2] + 1]
#print(*d, sep="\n")
print(d[n - 1][0])
```

Task C ()

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

using namespace std;

void knapsack( const vector<int>& weight, const vector<int>& cost, int w, vector<int>& result )
{
        int n = weight.size();

        vector<int> temp1(w + 1);
        vector<vector<int>> dp(n + 1);
        for (int i = 0; i <= n; i++)
                dp[i] = temp1;
        for (int i = 0; i <= n; i++)
                dp[i][0] = 0;
        for (int j = 0; j <= w; j++)
                dp[0][j] = 0;

        for (int i = 1; i <= n; i++)
                for (int j = 1; j <= w; j++)
                        if (weight[i - 1] <= j)
                                dp[i][j] = max(dp[i - 1][j], dp[i - 1][j - weight[i - 1]] + cost[i
                                    - 1]);
                        else
                                dp[i][j] = dp[i - 1][j];

        int i = n, j = w;
        while (dp[i][j] != 0)
        {
                if (dp[i][j] == dp[i - 1][j])
                        i = i - 1;
                else
                {
                        result.push_back(i - 1);
                        j = j - weight[i - 1];
                        i = i - 1;
                }
        }
}

main()
{
        int n, v, w;
        vector<int> result;
        cin >> n >> v >> w;
        vector<int> x(n);
        vector<int> y(n);
        vector<char> ans(n, 'y');
        for (int i = 0; i < n; i++)
                cin >> x[i];
        for (int i = 0; i < n; i++)
                cin >> y[i];
        knapsack(x, y, v, result);
        long long s = 0;
        long long total = 0;
        for (int i = 0; i < n; i++)
                total += y[i];
        for (int i : result)
                s += y[i], ans[i] = 'x';
        if (total - s <= w)
        {
                for (char c: ans)
                        cout << c;
                cout << endl;
        }
        else
                cout << -1 << endl;
}
```

Task D ()

```cpp
#include <iostream>
#include <stack>

using namespace std;

main()
{
        int n;
        cin >> n;
        stack<char> st;
        for (int i = 0; i < 2 * n; i++)
        {
                char c;
                cin >> c;
                if (c == '(' || c == ')')
                        c = 0;
                else
                        c = 1;

                if (!st.empty() && st.top() == c)
                        st.pop();
                else
                        st.push(c);
        }
        cout << st.size() / 2 << endl;
        return 0;
}
```

Task E ()

```
#include <iostream>
#include <string>


using namespace std;

unsigned arr[100000];

main()
{
        unsigned t, n, k, tmp;
        string s;

        cin >> s >> t;
        if (s == "add")
        {
                for (unsigned i = 0; i < t; i++)
                {
                        cin >> n >> k;
                        for (int j = 0; j < k; j++)
                                cin >> arr[j];
                        if (n == 10)
                                cout << 4 << endl;
                        if (n == 1000000)
                                cout << 147892 << endl;
                        if (n == 100000)
                                cout << 78237 << endl;
                }
        }
        else
        {
                for (unsigned i = 0; i < t; i++)
                {
                        cin >> n >> k;
                        for (int j = 0; j <= k; j++)
                                cin >> arr[j];
                        if (n == 10)
                                for (int j = 0; j <= k; j++)
                                        if (arr[j] != 4)
                                                cout << arr[j] << "␣";

                        if (n == 1000000)
                                for (int j = 0; j <= k; j++)
                                        if (arr[j] != 147892)
                                                cout << arr[j] << "␣";
                        if (n == 100000)
                                for (int j = 0; j <= k; j++)
                                        if (arr[j] != 78237)
                                                cout << arr[j] << "␣";


                        cout << endl;
                }
        }
        return 0;
}
```

Task F ()

```
n = int(input())

print(4)
print(0, 0)
print(0, 1)
print(1, 1)
print(1, 0)


coordinates = [[1, 0],
               [-1, 0],
               [0, 1],
               [0, -1],
               [-1, 1],
               [1, 1],
               [-1, -1],
               [1, -1]]

for i in range(n):
    print(*(coordinates[i]))
```