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

Task A ()

```cpp
#include <iostream>
#include <cmath>
#include <string>

using namespace std;

int main() {
    int a;
    string per = "023456789";
    cin >> a;
    if(a<10) {
        cout << a%10;
    }else{
        a-=10;
        cout << per[a%9];
    }
    return 0;
}
```

Task B ()

```cpp
#include <iostream>
#include <cmath>
#include <string>
#include <set>

using namespace std;

int main() {
    int n,k;
    cin >> n >> k;
    string s;
    cin >> s;
    int ans = 1;
    set<char> let;
    int i = 0;
    for(auto c : s){
        let.insert(c);
        i++;
        if(let.size()==4 || i>k){
            let.clear();
            i = 1;
            ans++;
            let.insert(c);
        }
    }
    cout << ans;
    return 0;
}
```

Task C ()

```cpp
#include <iostream>
#include <cmath>
#include <string>
#include <vector>
#include <bitset>

using namespace std;

int main() {
    int n,x,y;
    cin >> n >> x >> y;
    vector<pair<int,int>> o(n);
    int W = 0;
    for(int i = 0; i< n;i++) cin >> o[i].first;
    for(int i = 0; i< n;i++) { cin >> o[i].second; W+=o[i].second; }
    vector<int> pos(x+1,-1);
    vector<bitset<500>> who(x+1);
    pos[0] = 0;
    int p = 0;
    for(int i = 0; i < n;i++){
//        cout << i;
        for(int j = x - o[i].first; j > -1; j--){
            if(pos[j] == -1) continue;
            if(pos[j+o[i].first]< pos[j] + o[i].second){
                pos[j+o[i].first] = pos[j] + o[i].second;
                who[j+o[i].first] = who[j];
                who[j+o[i].first][i] = true;
                if (pos[p] < pos[j+o[i].first]){
                    p = j+o[i].first;
                }
            }
        }
    }
//        for(auto c : pos){
//            cout << c <<' ';
//        }
//        cout << '\n';
    }
//    cout << p<<'\n';
    if(W-pos[p] >y){
        cout << -1;
    }else{

//        for(int i = 0; i < x+1;i++) cout << parent[i]<<' ';
        for(int i = 0; i< n; i++) cout <<(who[p][i]?'x':'y');
    }
    return 0;
}
```

Task D ()

```cpp
#include <iostream>
#include <cmath>
#include <string>
#include <vector>
#include <stack>
using namespace std;

int main() {
    int n;
    cin >> n;
    string s;
    cin >> s;
    vector<bool> t(2*n);
    for(int i = 0; i< 2*n;i++){
        t[i] = (s[i]=='(' || s[i] == ')');
    }
    stack<bool> q;
    int i;
    for(i = 0; i< 2*n-q.size();i++){
        if(!q.empty() && q.top() == t[i]) q.pop();
        else q.push(t[i]);
    }
    int ans = 0;
    for(i; i < 2*n;i++){
        ans += q.top() != t[i];
        q.pop();
    }
    cout << ans;
    return 0;
}
```

Task E ()

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

using namespace std;

void add(){
    int n,k;
    cin >> n >> k;
    vector<int> a(k);
    for(int i = 0; i< k;i++) cin >> a[i];
    sort(a.begin(),a.end());
    int prev = 0;
    for(auto c : a){
//          cout << c << ' '<<prev <<'\\'<<'\n';
        if(c-prev > 3) {
            cout << prev+1 << '\n';
            return;
        }
        prev = c;
    }
    cout << a.back() + 1<<'\n';
    return;
}

void clear(){
    int n,k;
    cin >> n >> k;
    vector<int> a(k+1);
    for(int i = 0; i< k+1;i++) cin >> a[i];
    sort(a.begin(),a.end());
    int ans = a[k];
    for(int i = 0; i< k; i++){
        if(a[i+1] - a[i] > 2){
            ans = a[i];
            break;
        }
    }
    for(int i = 0;  i< k+1; i++){
        if(a[i] != ans) cout << a[i]<<'␣';
    }
    cout <<'\n';
    return;
}

int main() {
    string s;
    cin >> s;
    int t;
    cin >> t;
    while (t--){
        if(s == "add") add();
        else clear();
    }
    return 0;
}
```

Task F ()

```cpp
#include <iostream>
#include <cmath>
#include <string>
#include <vector>
#include <bitset>

using namespace std;

int main() {
    int n;
    cin >> n;
    if(n<9){
        cout << 4<<'\n';
        cout <<"0_0\n0_1\n1_1\n1_0\n";
        int ans = 0;
        for(int x = -1; x < 2; x++){
            if(ans == n) break;
            for(int y = -1; y<2;y++){
                if(x == y && x == 0) continue;
                if(ans == n) break;
                cout << x <<'_'<< y <<'\n';
                ans++;
            }
        }
    }
    return 0;
}
```