

```

#include <iostream>
#include <fstream>

using namespace std;

struct Lancio {
    int risultato;
    Lancio *pun;
};

struct LancioDadi {
    Lancio *testa;
    int N;
    int quanti;
};

void inizializzaLancioDadi(LancioDadi &LD, int N) {
    if (N <= 0)
        LD.N = 20;
    else
        LD.N = N;

    LD.quanti = 0;
    LD.testa = NULL;
}

void inserisciLancio(LancioDadi &LD, int x) {
    if (x < 2 || x > 12)
        return;

    if (LD.quanti == LD.N) {
        //Se ho già N lanci, elimino quello in coda prima
        Lancio *q, *p;
        for (q = LD.testa; q->pun != NULL; q = q->pun)

```

```

        p = q;

        p->pun = NULL;
        delete q;

        LD.quantiti--;
    }

    //Inserisco in testa l'ultimo risultato, a prescindere
    Lancio *r = new Lancio;
    r->risultato = x;
    r->pun = LD.testa;
    LD.testa = r;
    LD.quantiti++;
}

double calcolaFrequenza(LancioDadi &LD, int x) {

    if (LD.testa == NULL)
        return 0.0;

    double conta = 0.0;

    for (Lancio *q = LD.testa; q != NULL; q = q->pun)
        if (q->risultato == x)
            conta++;

    return conta / LD.quantiti;
}

void inserisci_in_fondo(Lancio *&testa, int x) {

    Lancio *p, *q;

```

```

    for (q = testa; q != NULL; q = q->pun)
        p = q;

    Lancio *r = new Lancio;
    r->risultato = x;

    r->pun = NULL;
    if (testa == NULL)
        testa = r;
    else
        p->pun = r;
}

Lancio *sottoLista(LancioDadi &LD, int k) {

    if (k < 2)
        return NULL;

    Lancio *testa = NULL;

    for (Lancio *q = LD.testa; q != NULL; q = q->pun)
        if (q->risultato <= k)
            inserisci_in_fondo(testa, q->risultato);

    return testa;
}

void stampaLancioDadi(LancioDadi &LD) {

    cout << '[' << LD.N << ", " << LD.quantita << "]: ";

    cout << '<';

    for (Lancio *q = LD.testa; q != NULL; q = q->pun) {

```

```

        cout << q->risultato;
        if (q->pun != NULL)
            cout << ", ";
    }

    cout << '>' << endl;
}

int *arrayLanci(LancioDadi &LD) {

    int *v = new int[LD.N];

    for (int i = 0; i < LD.N; i++)
        v[i] = 0;

    int i = 0;
    for (Lancio *q = LD.testa; q != NULL; q = q->pun) {
        v[i] = q->risultato;
        i++;
    }

    return v;
}

void f2(const char* output, int N) {

    ofstream out;

    out.open(output);

    if(!out)
        return;

    for(int i=1;i<=N;i++) {

```

```

        for (int j = 0; j < i; j++)
            out << '★';
        out << endl;
    }

    out.close();
}

int f3(int *v, int N, int k) {
    if (N==0)
        return 0;

    if (v[N-1] <= k)
        return 1+f3(v, N-1, k);
    else
        return 0+f3(v, N-1, k);
}

int main() {
    LancioDadi LD;
    inizializzaLancioDadi(LD, 20);
    stampaLancioDadi(LD);                // [20, 0]: < >

    inserisciLancio(LD, 5);
    inserisciLancio(LD, 7);
    inserisciLancio(LD, 2);
    inserisciLancio(LD, 12);
    inserisciLancio(LD, 7);
    stampaLancioDadi(LD);                // [20, 5]: <7, 12, 2, 7, 5>

    double d;

```

```

d = calcolaFrequenza(LD, 7);    // d = 0.4

cout << "Frequenza di 7: " << d << endl;

inserisciLancio(LD, 3);
stampaLancioDadi(LD);          // [20, 6]: <3, 7, 12, 2, 7, 5>

d = calcolaFrequenza(LD, 7);    // d = 0.3333
cout << "Frequenza di 7: " << d << endl;

int *v = arrayLanci(LD);

cout << '[' << v[0];
for (int i = 1; i < 20; i++) {
    cout << ", " << v[i];
}
cout << ']' << endl;

Lancio *testa = sottoLista(LD, 6);

for (Lancio *q = testa; q != NULL; q = q->pun) {
    cout << q->risultato;
    if (q->pun != NULL)
        cout << "->";
}

cout << endl;

for (int i = 0; i < 20; i++) {
    int x = rand() % 11 + 2;
    cout << "inserisco " << x << endl;
    inserisciLancio(LD, x);
}

```

```
    stampaLancioDadi(LD);

    f2("output.txt", 15);

    int array[11] = {1, 5, 4, 3, 6, 9, 4, 5, 6, 6, 8};

    int n = f3(array, 11, 5);

    cout << "L'array di interi ha " << n << " valori minori di 5" << endl;

}
```

```
/* Esercizio 1:
* la conversione in base 10 del numero 10010101 in complemento a due e' -107
* la conversione del numero -107 in base due modulo e segno e' 11101011
*/
```