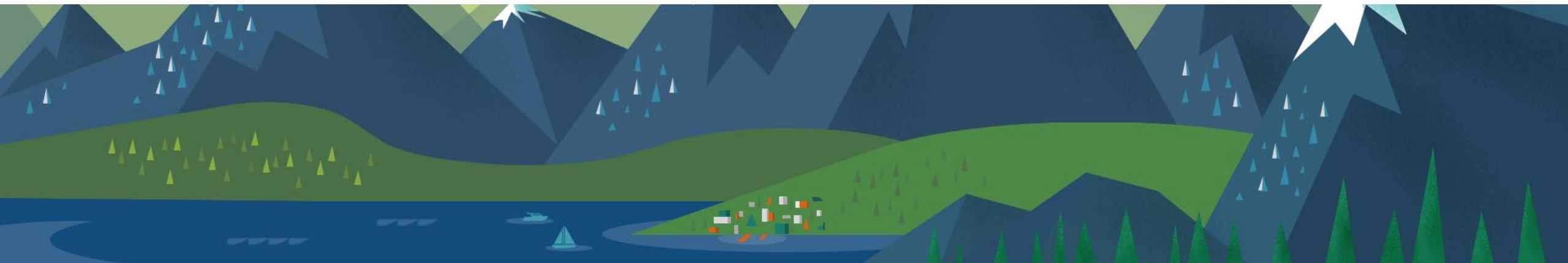




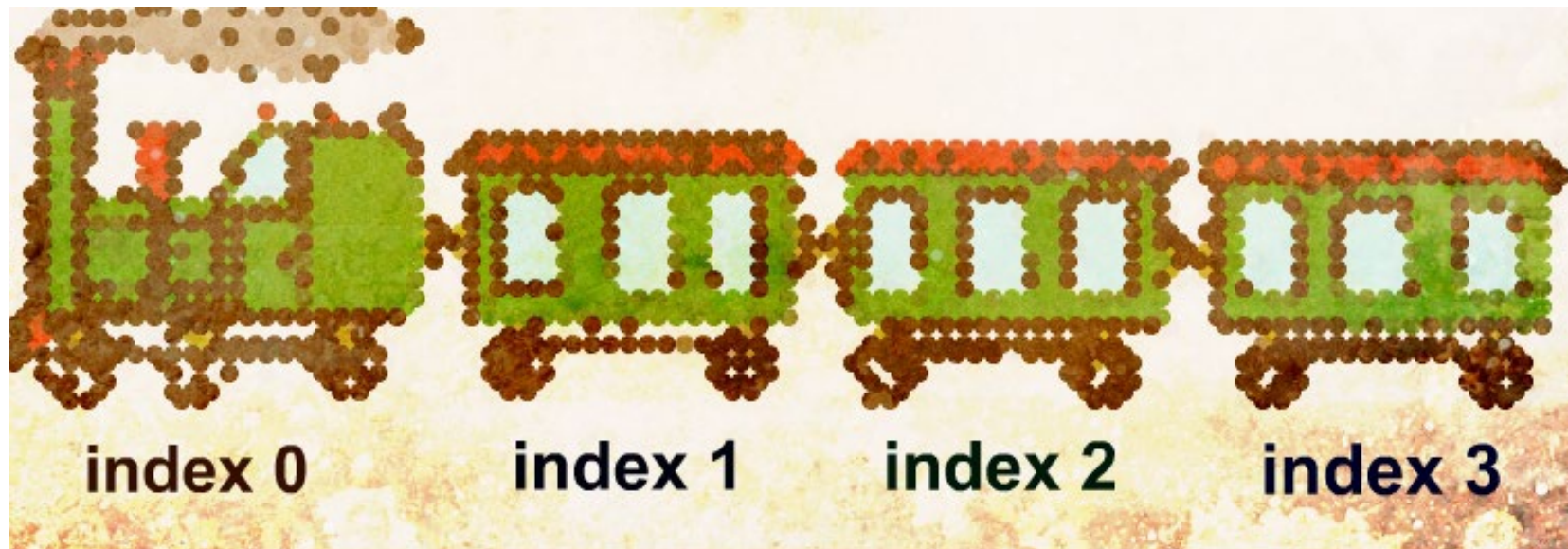
C++da massivlar



Amaliy mashg'ulotning maqsadi

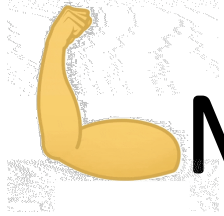
- C++da massivlar bilan ishlash ko'nikmasini shakilantirish, massivlarda ketma-ket va alohida ma'lumotlarni ko'rsatishni o'rganish.

Massiv?



Tassavur qilamiz siz bilan 500ga yaqin oz'garuvchilarni hosil qilishimiz va dasturning ayrim hollarida ularni ishlatishimiz kerak bunda **{int,short,long,bool,double,char}** kengaytmasi ishlatiladi.

Bu holda **{int,short,long,bool,double,char}** tipiga mansub barcha o'zgaruvchilarni bitta guruhga keltirish va u guruhdan alohida bitta o'zgaruchini belgilash imkoni massiv orqali beriladi.



Massivning asosiy imkoniyatlari

- C++ tilidagi massiv - bu bir xil nomga ega bo'lgan bir xil turdagi o'zgaruvchilarning ma'lum sonining yig'indisidir. Masalan, `int array[3]`; Bu yozuv biz int tipidagi 3 ta o'zgaruvchini o'z ichiga olgan array deb nomlangan massivni e'lon qilganimizni bildiradi;
- massiv o'zgaruvchilari **elementlar** deb ataladi; har bir element o'zining noyob indeksiga ega - uning seriya raqami. Indeksdan foydalanib, biz ma'lum bir elementga murojaat qilishimiz mumkin.
- **MUHIM!** - massiv elementlarini indekslash 0 dan boshlanadi. Shunday qilib, massiv int massivida [3] birinchi element 0 indeksga, oxirgi element esa 2 indeksga ega. Masalan, massivning nol elementiga murojaat qilish va uning qiymatini o'zgartirish uchun siz massiv nomini va kvadrat qavs ichida element indeksini ko'rsatishingiz kerak - `massiv[0] = 33`.



```
1 // size hajmiga ega massiv yaratamiz,  
2 // for sikli orqali kataklarga ma'lumot kirg'izamiz  
3 // va barcha ma'lumotlarni massivda aks ettiramiz  
4  
5 #include <iostream>  
6 using namespace std;  
7  
8 int main()  
9 {  
10  setlocale(LC_ALL, "eng");  
11  
12  const int SIZE = 10; //konstattani belgilaymiz  
13  int firstArray[SIZE ]; //massiv miqdorini belgilovchi SIZE kattaligini belgilaymiz  
14  
15  for (int i = 0; i < SIZE ;i++) //to'ldirilgan ma'lumotlarni ekranda aks ettiramiz  
16  {  
17    firstArray[i] = i + 0; // sikilning birinchi qadamida firstArray[0] 1 (0 + 1)  
        qiymati beriladi  
18    cout << i << "- chi katakdagi qiymat" << firstArray[i] << endl;  
19  }  
20  cout << endl;  
21  
22  return 0;  
23 }
```

```
/tmp/Zxf3aVTmHy.o  
-1- chi katakdagi qiymat-  
0- chi katakdagi qiymat0  
1- chi katakdagi qiymat1  
2- chi katakdagi qiymat2  
3- chi katakdagi qiymat3  
4- chi katakdagi qiymat4  
5- chi katakdagi qiymat5  
6- chi katakdagi qiymat6  
7- chi katakdagi qiymat7  
8- chi katakdagi qiymat8  
9- chi katakdagi qiymat9
```

Massivlarni aks ettirish usullari

```
int firstArray[5] = {11, 12, 13, 14, 15};
```

Shunday qilib, 0 - firstArray[0] indeksidagi elementga 11 qiymati, firstArray[4] massivining oxirgi elementiga esa 15 qiymati beriladi.

```
int firstArray[ ] = {11, 12, 13, 14, 15};
```

ikkinchi holda, kompilyator jingalak qavslardagi {} ma'lumotlar miqdoridan kelib chiqib, massiv hajmini avtomatik ravishda hisoblab chiqadi.

```
int firstArray[5] {};
```

C++11 da (kodlash standarti), ro'yxatni ishga tushirishdan (jingalak qavslar bilan boshlash) foydalanilganda, hatto = belgisini qoldirishga ruxsat beriladi.

Vazifa!

50 ta elementdan iborat massivni 1 dan 99 gacha toq raqamlar bilan to'ldiring. (raqamning juft yoki yo'qligini tekshirish uchun modul operatoridan foydalaning)

main.cpp



Run

Output

Clear

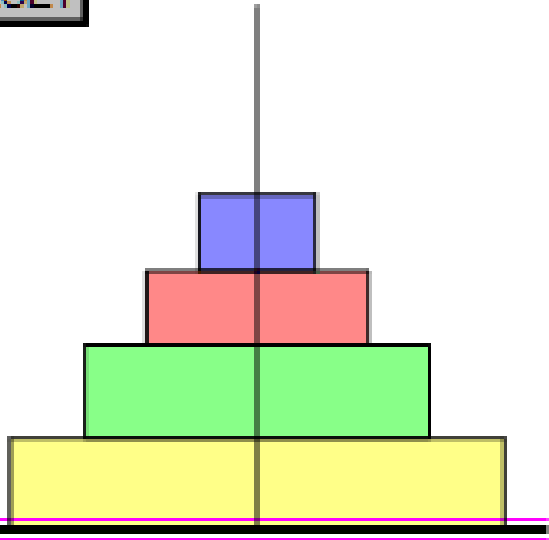
```
1 #include <iostream>
2 #include <ctime>
3 #include <cstdlib>
4 using namespace std;
5
6 int main()
7 {
8     setlocale(LC_ALL, "rus");
9
10    const int SIZE = 50;
11
12    int ourArr[SIZE] = {};
13
14    srand(time(NULL));
15
16    for( XXXXXXXXXXXXXXXX )
17    {
18        ourArr[i] = 1 + rand() % 100;
19        if(ourArr[i] % 2 != 0)
20        {
21            cout << ourArr[i] << " X ";
22            i++;
23        }
24    }
25    cout << endl << endl;
26    return 0;
27 }
```

/tmp/Zxf3aVTmHy.o

```
1 | 69 | 21 | 35 | 9 | 9 | 25 | 65 | 5 | 55 | 97
   | 95 | 25 | 23 | 89 | 89 | 17 | 59 | 85 | 79
   | 45 | 53 | 69 | 89 | 53 | 57 | 19 | 63 | 1
97 | 75 | 3 | 37 | 19 | 47 | 91 | 65 | 21 |
53 | 15 | 71 | 49 | 55 | 55 | 79 | 29 | 81 |
93 | 91 | 11 |
```

START

RESET



main.cpp

```
1 #include <stdlib.h>
2 #include <iostream>
3 using namespace std;
4
5 // =====> xalqalrani almashtirish sxemasi
6 // -|- | |
7 // --|-- | |
8 // ---|--- | |
9 // 1 2 3
10
11 int nopr = 0; // operatsiyalar hisoblagichi
12
13
14 void put( int from, int to ) {
15     cout << from << " => " << to << " | ";
16     if( ( XXXXXXXXXXXXXXXX ) cout << endl;
17 }
18 //from dan to n xalqasiga o'tkazish:
19 void move( int from, int to, int n ) {
20     int temp = from ^ to; // oraliq pozitsiya
21     if( 1 XXXXXXXXXXXXXXXX ); // bitta xalqani o'tkasiz
22     else {
```

```
23         move( from, temp, n - 1 ); // n-1 ni temp ga o'tkazish
24         put( from, to ); // to eng paski qiymatga o'tkazish
25         move( temp, to, n - 1 ); // n-1 ni temp dan to ga o'tkazish
26     }
27 }
28
29 int main( int argc, char **argv, char **envp ) {
30     int n = 5; //o'tkazilishi kerak bo'lgan xalqalar
31     soni
32     if( XXXXXXXXXXXXXXXX ) != 0 )
33         n = atoi( argv[ 1 ] );
34     cout << "piramida soni: " << n << endl;
35     move( 1, 3, n ); // yechim !
36     if( 0 != ( nopr % 5 ) ) cout << endl;
37     cout << "kombinatsiyalar soni " << nopr << endl;
38     return 0;
39 }
```