

Би өөрийгөө Программист, Хөгжүүлэгч, Инженерүүд дотроос дийлэнх хувь нь программист талдаа байгаа гэж бодож байгаа. Яагаад гэвэл өмнөх жилүүдэд программчлалын хэлүүдийг суралцаж тухайн хэлүүдийг ашиглаж бэлэн загвар, мөн баримт бичгийн дагуу программчилдаг байсан. Одоогоор ямар нэгэн системийг бүтээхийг зорьвол ямар 2 шаардлага шинжилгээ хийх болон зохиомж архитектураа гаргасны үндсэндээр системийг алдаагүй зөв хөгжүүлэх зорилготой ингэснээр хөгжүүлэгч болон гэсэн зорилготой.

1. Энэ юуны тухай рарег вэ?

- Бодлого 1
  - Уг бодлого нь хэрэглэгч функц болон үндсэн функцүүдээс тогтоно. Бодлогоны хувьд `int` буюу бүхэл тоон параметр дамжуулж функц нь тэр тоог давталтаар 1 хүртэл үргэлжлэх бөгөөд уг тоог тэгш бол 2-т хуваана сондгой бол 3-д үржээд нэгийг нэмэгдэнэ
- Бодлого 2
  - Бодлогоны хувьд массив төрлийн хувьсагч агуулсан параметртэй функц байна. Уг элементүүдээс 2,3,5 тоонуудад хуваагддаг мөн аль алинд нь хуваагдахгүй элементүүдийг тоолно. Үүнийг 2 аргаар бодож болох юм.
    - Алхам 1. Хуваагчийг массивийн элемент бүрийн утганд үлдэгдэлтэй хуваагдаж байх эсэхийг тус бүрт нь давталт ашиглан тоолж болно
    - Алхам 2. Тус бүрт нь хувьсагч оноогоод нэг давталтаар нөхцөл шалгаж тоолж болно болно.

2. Энэ ямар архитектур вэ?

- Энэ нь бие даасан хэрэглэгч функцүүдтэй бөгөөд өөр өөрийн гэсэн бие даасан хэрэгжүүлэлт хийдэг байна

Бодлогууд-н хувьд `lengthBumpy(int a)`, `is235Array(int[] a)` хэрэглэгч функц нь параметрээр ирсэн утгыг хүлээж аваад хэрэгжүүлэх ажил гүйцэтгэнэ.

- `lengthBumpy(int a)` нь дотроо параметрээр орж ирсэн утгыг давталтаар тэгш байнуу сондгой байна уу гэдгээс хамаарч үйлдэл хийсээр 1 дээр ирээд зогсоно.  
**Тэгш( $n=n/2$ ), сондгой( $n=n*3+1$ )**
- `is235Array(int[] a)` энэ нь параметрээр орж ирсэн элементүүдийг аваад хуваагдагч тус бүрт нь давталтаар гүйлгэж тоолоод массивийн урттай харьцуулж үнэн, худал утга буцаадаг байх

3. Яаж сайжруулсан?

• **Бодлого 1**

```

23 public static void main(String[] args) {
24     lengthBumpy(50);
25 }
26 }
27
input
50 25 76 38 19 58 29 88 44 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
25

21 else {System.out.println("Buruu utga orj irlee");return 0;}
22 }
23 public static void main(String[] args) {
24     lengthBumpy(-2);
25 }
26 }
27
input
Buruu utga orj irlee

```

```

public class Main
{
    public static int lengthBumpy(int n){
        if(n>0){
            int s = 1;
            System.out.print(n+" ");
            for(int i = n;i>1;){
                if(i%2==0){
                    i = i/2;
                }
                else{
                    i = i*3+1;
                }
                s++;
                System.out.print(i+" ");
            }
            System.out.println();
            System.out.print(s+" ");
            return s;
        }
        else {System.out.println("Buruu utga orj irlee");return 0;}
    }

    public static void main(String[] args) {
        lengthBumpy(50);
    }
}

```

}

## • Бодлого 2

○ Өмнөх нь

FALSE

```
54 public static void main(String[] args) {  
55     int[] a = {1,2,3,4,6};  
56     System.out.print("\n"+is235Array(a));  
57 }  
58 }  
59
```

input

```
3+2+0+1=6  
6~5  
false
```

Program finished with exit code 0

TRUE

```
54 public static void main(String[] args) {  
55     int[] a = {1,2,3,4,5};  
56     System.out.print("\n"+is235Array(a));  
57 }  
58 }  
59
```

input

```
2+1+1+1=5  
5~5  
true
```

// !!!! Niit 4n davtalt ashglasn ygd 1 davtalt ashglaaguive gevel

//input: {1,2,3,4,5}

//iim massiv bailaa gehed

//2t huvaagddag {0+1+0+1+0} 2

//3t huvaagddag {0+0+1+0+0} 1

//5t huvaagddag {0+0+0+0+1} 1

//235t zereg huvaagdahgui {1+0+0+0+0} 1

//output: 2+1+1+1

//ehleed niit massivaas 2d huvaagdah massiv-g toolno. Duussanii daraa ene shig 3,5d huvaagdah index-uudiig neg burchlen toolno

```
//4 deh davtalt ni ene 3ni toond zereg huvaagdahgui indexiig
public class Main
{
    public static boolean is235Array(int a[]) {
        int i,//davtaltiin huvisagch
        s = 0,
        view = 0;
        for(i = 0,view = 0;i<a.length;i++){
            if(a[i]%2==0){
                s++;
                view++;
            }
        }
        System.out.print(view);
        for(i = 0,view = 0;i<a.length;i++){
            if(a[i]%3==0){
                s++;
                view++;
            }
        }
        System.out.print(" "+view);
        for(i = 0,view = 0;i<a.length;i++){
            if(a[i]%5==0){
                s++;
                view++;
            }
        }
        System.out.print(" "+view);
        for(i = 0,view = 0;i<a.length;i++){
            if(a[i]%2!=0 && a[i]%3!=0 && a[i]%5!=0){
                s++;
                view++;
            }
        }
        System.out.print(" "+view+"=");
        System.out.println(s);

        System.out.print(s+"~"+a.length);
        if(s==a.length){
            return true;
        }
        return false;
    }

    public static void main(String[] args) {
```

```
        int[] a = {1,2,3,4,5};
        System.out.print("\n"+is235Array(a));
    }
}
```

## • Сайжруулсан

**Давталтыг нэг болгоод two, three, five other гэсэн хувьсагчид тус тусад нь нэмж өгчөөд үр дүндээ нэгтгэж хэвэлнэ**

```
public class Main
{
    public static boolean is235Array(int a[]) {
        int i, //davtaltiin huvisagch
        s = 0, two = 0, three = 0, five = 0, other = 0;
        for(i = 0; i < a.length; i++){
            if(a[i] % 2 == 0){
                two++;
            }
            if(a[i] % 3 == 0){
                three++;
            }
            if(a[i] % 5 == 0){
                five++;
            }
            if(a[i] % 2 != 0 && a[i] % 3 != 0 && a[i] % 5 != 0){
                other++;
            }
        }
        int cout = two + three + five + other;

        System.out.print(two + " " + three + " " + five + " " + other + " => " + cout + "\n" + cout + "~" + a.length);
        if(cout == a.length){
```

```
        return true;
    }
    return false;
}

public static void main(String[] args) {
    int[] a = {1,2,3,4,5};
    System.out.print("\n"+is235Array(a));
}
}
```

- Функцийг ашиглах зорилго нь кодыг маш эмх цэгцтэй хурдыг сайжруулах зорилгоор ашигладаг. Олон удаа ашиглах процессийг функцэнд хийж өгчөөд хэрэгтэй газраа дахин ашиглахад давуу талтай