

Лабораторийн ажил 5: Стандарт классын сан ашиглах

Лабораторийн ажлын зорилго:

Рекурс функц бичих, Math, String, Date классуудыг ашиглах.

Лабораторийн ажлын суралцахуйн үр дүнгүүд:

Энэ лабораторийн ажлыг гүйцэтгэснээр оюутан дараах чадваруудтай болсон байна.

д/д	Суралцахуйн үр дүнгүүд	Суралцахуйн үр дүнг илэрхийлэх үйл үг	Суралцахуйн үр дүнгийн түвшин (Блумын)	CLOs хамаарал
1	Тогтмол ашиглах	Ашиглах (Use)	Хэрэглээ	6
2	Бэлэн классын үйлдлүүд ашиглах	Ашиглах (Use)	Хэрэглээ	6
3	Тэмдэгт мөр ашиглах	Ашиглах (Use)	Хэрэглээ	6
4	Тайлан хамгаалах	Хамгаалах (defend)	Синтез	12
5	Англи хэл дээр холбогдох материал бусад эх үүсвэрээс унших	Унших (Read)	Ойлголт	13

Ашиглах програм хангамж/техник хангамж, бусад хэрэглэгдэхүүнүүд:

Лабораторийн компьютер эсвэл өөрийн зөөврийн компьютерийг ашиглана.

Онолын ойлголтууд:

Рекурс функц

Өөр дотроо өөрийгөө дууддаг функцийг рекурс функц гэнэ. Рекурс функцийг програмчлалд олон төрлийн бодлогыг хялбар аргаар бодоход ашигладаг. Жишээлбэл, тооны факториаль олох бодлогыг рекурс функц ашиглан хялбархан бодож болно. Факториаль олох функц нь:

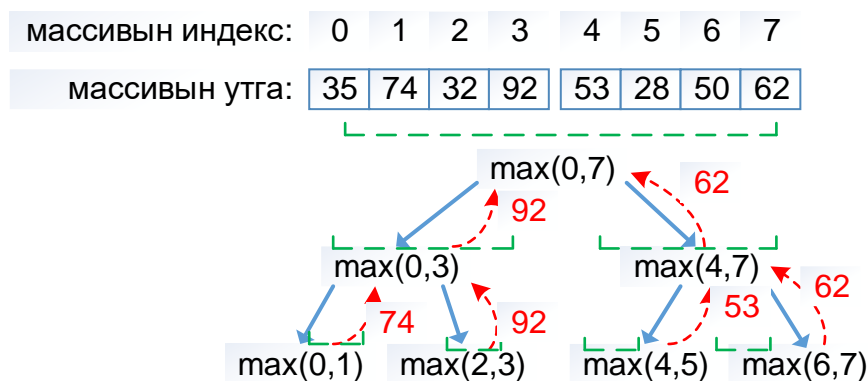
- $F(n)=n \cdot F(n-1)$ гэсэн рекурент томъёогоор илэрхийлэгдэнэ,
- Мөн $F(n)=1$ байдаг.

Иймээс факториаль олох `fac(int n)` функцийг дараах байдлаар бичиж болно.

```
static int fac(int n)
{
    if ( n == 1 )
        return 1;
    else
        return n * fac(n-1);
}
```

Дээрх жишээнээс харахад рекурс функцэд өөрөө өөрийгөө дуудах үйл ажиллагааг зогсоох хязгаарын нөхцөл байх шаардлагатай байна. Дээрх жишээнд `n==1` нөхцөл нь энэ нөхцөл болж байна. Хэрэв ийм нөхцөл байхгүй бол рекурс функц хязгааргүй өөрийгөө дуудаж програм гацна.

Рекурсивная функция может быть разделена на части (divide and conquer) и решена с помощью рекурсии. Например, массив элементов может быть разделен на две части, и для каждой части может быть найден максимальный элемент. Этот процесс может быть повторен для каждой части, пока не останется один элемент. Тогда можно будет сравнить элементы и найти максимальный. Этот процесс может быть повторен для каждой части, пока не останется один элемент. Тогда можно будет сравнить элементы и найти максимальный. Этот процесс может быть повторен для каждой части, пока не останется один элемент. Тогда можно будет сравнить элементы и найти максимальный.



Зураг 5.1 Массивын элементүүдийг эрэмбэлэх

Дээрх жишээнд 8 элементтэй массивын хязгаар 0-ээс 7 байна. Үүнийг 0-оос 3, 4-өөс 7 гэж хуваав. 0-ээс 3 хязгаарыг 0-ээс 1, 2-оос 3 гэж хуваав. Эдгээр хязгааруудаас тус тус 74, 92 гэсэн ХИ утга буцна. Эдгээрийг харьцуулж 92 гэсэн утга буцна гэх мэтчилэн ХИ утгыг олж болно. Энэ алгоритмыг Жава хэл дээр илэрхийлбэл дараах хэлбэртэй болно.

```
public class MaxRecurse
{
    public static void main(String[] args)
    {
        int[] a = {35, 74, 32, 92, 53, 28, 50, 62};
        System.out.println("Max: " + max(0, 7, a));
    }
    public static int max( int i, int j, int[] arr)
    {
        if ( (j - i) <= 1)
        {
            if (arr[j] >= arr[i])
                return arr[j];
            else
                return arr[i];
        }
        else
        {
            int m1 = max(i, (i+j)/2, arr);
            int m2 = max((i+j)/2+1, j, arr);
            if (m1 >= m2)
                return m1;
            else
                return m2;
        }
    }
}
```

}

Math класс

Math класс нь математикийн олон төрлийн функцуудыг агуулдаг. **java.lang** пакетэд агуулагдана.

Math классын зарим функцүүдтэй танилцъя. Үүнд:

1. `Math.abs(n)` — n тооны модуль буцаах.
2. `Math.round(n)` — n тоотой хамгийн ойр бүхэл тоон утгыг буцаах буюу n тоог бүхэл болгох.
3. `Math.ceil(n)` — n тоотой хамгийн ойр, таслалаас хойш орон 0 байх утгыг буцаах. Жишээ нь: `Math.ceil(3.4)` буцаах утга нь (4.0).
4. `Math.cos(n)`, `Math.sin(n)`, `Math.tan(n)` — n радианы \sin , \cos , \tan зэрэг функцуудын утгыг олох.
5. `Math.acos(n)`, `Math.asin(n)`, `Math.atan(n)` — урвуу тригонометр функцүүд өнцгийг радианаар буцаах.
6. `Math.sqrt(n)` — n -ээс квадрат язгуур авах.
7. `Math.pow(n, b)` — n тоог b зэрэгт дэвшүүлэх.
8. `Math.log(n)` — n тооноос натураль логарифм авах.
9. `Math.log10(n)` — n тооноос 10 суурьтай логарифм авах.

Тогтмол тоо:

1. `Math.PI` — «пи» тоо, таслалаас хойш 15 тэмдэгтийн нарийвчлалтай авна.
2. `Math.E` — e тоо (экспоненциаль функцийн суурь), таслалаас хойш 15 тэмдэгтийн нарийвчлалтай авна.

Жишээлбэл:

```
System.out.println(Math.abs(-2.33)); // 2.33
System.out.println(Math.round(Math.PI)); // 3
System.out.println(Math.round(9.5)); // 10
System.out.println(Math.round(9.5-0.001)); // 9
System.out.println(Math.ceil(9.4)); // 10.0
double c = Math.sqrt(3*3 + 4*4);
System.out.println(c); // 3 ба 4 талуудтай тэгш өнцөгт гурвалжны гипотенузын утга
double s1 = Math.cos(Math.toRadians(60));
System.out.println(s1); // 60 градусын косинусын утга
```

Санамсаргүй тоо

1. `Math.random()` функц нь $[0;1)$ завсарт орших бодит санамсаргүй утгыг буцаадаг. Жишээлбэл, дараах жишээ нь k бүхэл тоон хувьсагчид 0-ээс 10 хүртэлх санамсаргүй бүхэл тоог олгоно.

```
int k = (int) (Math.random()*10);
```

Жишээ:

```

System.out.println(Math.random()); // тоо [0;1)завсраас буцаана
System.out.println(Math.random()+3); // тоо [3;4) завсраас
System.out.println(Math.random()*5); // тоо [0;5) завсраас
System.out.println( (int) (Math.random()*5) ); // бүхэл тоо [0;4]
завсраас
System.out.println(Math.random()*5+3); // тоо [3;8) завсраас
System.out.println( (int) (Math.random()*5+3) ); // бүхэл тоо [3;7]
завсраас
System.out.println( (int) (Math.random()*11) - 5); //бүхэл тоо [-5;5]
завсраас

```

2. Random классын nextInt(n) метод нь 0 -оос n-1 хүртэл санамсаргүй тоо үүсгэнэ.

Жишээ:

```

import java.util.Random;

. . .

Random random = new Random();

int number = random.nextInt(11); //return x, 0 <= x <= 10

```

max-аас min хүртэлх санамсаргүй бүхэл тоо үүсгэх бол:

```

int number = random.nextInt(max - min + 1) + min;

```

char тэмдэгт төрөл

Жава хэлэнд тэмдэгтийг char өгөгдлийн төрөлтэй зарладаг. Тэмдэгт хувьсагчид утга олгохдоо ‘ ‘ хаалт ашиглана. Жишээлбэл, тэмдэгт төрлийн хувьсагч нь таслал илэрхийлбэл:

```

char comma = ' , ' ;

```

Тэмдэгт бүр давтагдашгүй тоон утгатай байдаг. Үүнийг *үсгийн код* гэнэ. Латин үсгүүд болон цифрүүд кодын хүснэгтэд дараалан байрладаг. Үүнийг батлахын тулд дараах жишээг үзүүлэв. Үүнд ‘a’ үсгийн дараа ‘b’ үсэг байдаг болохыг харуулав.

```

char ch = 'a';
ch++;
System.out.println(ch);

```

Дэлгэцэд b үсэг хэвлэгдэнэ.

Мөн үсэг бүрд харгалзах кодыг дараах байдлаар дэлгэцэд хэвлэж болно. Дараах жишээнд том A үсэг болон жижиг a үсгийн кодыг хэвлэнэ.

```

System.out.println((int)'A' + " ; " + ((int) 'a'));

```

Дэлгэцэд 65; 97 гэж хэвлэгдэнэ.

Тэмдэгтүүдийн кодыг харьцуулж болно.

```

char ch = 'g';
if (ch > 'a')
    System.out.println("YES");

```

Дэлгэцэд YES гэж хэвлэнэ.

Дараах жишээнд тэмдэгт нь цифр (0-оос 9 хүртэл тоо) мөн эсэхийг шалгаж байна.

```

if (ch >= '0' && ch <= '9')
    // ...

```

Хэрэв тэмдэгт нь цифр байвал түүнийг тоон утга руу шилжүүлэх шаардлага гардаг. Дараах жишээ нь тэмдэгтээс цифр(тоо) рүү хөрвүүлж байна.

```
int k = ch - '0';
```

Энд '0' тэмдэгтийн код нь 48 бөгөөд 1,2,3 гэх мэт тоонуудын цифр 49,50,51 гэх мэт дараалласан кодтой байдаг. Эсрэгээр нь цифрээс тэмдэгт үүсгэх нь дараах хэлбэртэй байна.

```
char ch = (char) (k + '0');
```

Character класс

Тэмдэгтийн кодыг санах шаардлагагүй, Character классын функцүүдтэй ажиллах боломжтой.

Байгуулагч функц

```
Character(char c)
```

Авах утга:

```
char charValue()
```

Тэмдэгтийг цифр рүү хөрвүүлэх (radix–тооллын системийн суурь, ихэвчлэн 10-тх байна)

```
static int digit (char ch, int radix)
```

Эсрэг хөрвүүлэлт:

```
char fordigit (int digit, int radix)
```

Тэмдэгт нь үсэг мөн эсэх:

```
boolean isLetter()
```

Тэмдэгтийг том үсэг болгох:

```
char toUpperCase (char ch)
```

Character классын жишээ:

```
char c1 = '3';
```

```
// digit статик функц дуудалт
```

```
int d = Character.digit(c1, 10);
```

```
System.out.println("Square of " + c1 + " is " + d * d);
```

```
char c2 = 'a';
```

```
// ch объект үүсгэх
```

```
Character ch = new Character(c2);
```

```
// цаашид ch объектоор ажиллана
```

```
if (Character.isLetter(c2))
```

```
System.out.println(c2 + " is letter");
```

```
System.out.println("Capitalized " + ch.charValue() + " is " +
```

```
Character.toUpperCase(c2));
```

Дэлгэцэд хэвлэгдэх үр дүн:

Squareof 3 is 9
a is letter
Capitalized a is A

String класс

Тэмдэгт мөр нь тэмдэгтүүдийн дараалал юм. String тэмдэгт мөр нь String классаас үүсэх бөгөөд тэмдэгтүүдийн дарааллыг өөртөө агуулдаг.

String классаас тэмдэгт мөрийг хэд хэдэн янзаар үүсгэж болно. Дараах жишээнд хоёр янзаар ижил утга бүхий тэмдэгт мөрүүдийг үүсгэсэн байна.

```
String str = new String ("abc");  
String str1 = "abc";
```

Дараах жишээнд тэмдэгт массиваас тэмдэгт мөр үүсгэсэн байна.

```
char[] chs = {'a', 'b', 'c', 'e', 'z', 'k'};  
String str = new String(chs);
```

Тэмдэгт мөрүүдийг '+' оператор ашиглан залгаж болно.

```
String address = "Baga toiruu 29-36";  
String str1 = "The hotel address is" + address;  
String str2 = "new";  
str2+="student";
```

Үүнд:

- 1 + 23 ➡ Үр дүн 24
- "1" + "23" ➡ Үр дүн 123
- 1 + Integer.parseInt("23") ➡ Үр дүн 24.

String классын үйлдлүүд

String concat(String str)	Тэмдэгт мөрийг залгах функц
int length()	Тэмдэгт мөрийн уртыг олох функц
String substring(int beginIndex, int endIndex)	Тухайн тэмдэгт мөрөөс дэд тэмдэгт мөр авах функц
char charAt(int index)	Тэмдэгт мөрөөс нэг тэмдэгт авах функц
void getChars(int srcBegin, int srcEnd, char[] dst, int dstBegin)	Тэмдэгт мөрөөс хэд хэдэн тэмдэгт авах функц
boolean equals(Object anObject)	Тэмдэгт мөрүүд тэнцүү эсэхийг шалгах функц
boolean equalsIgnoreCase(String anotherString)	Тэмдэгт мөрүүд тэнцүү эсэхийг шалгах функц (үсэгний том, жижгийг харгалзахгүй)
boolean startsWith(String prefix)	Тодорхойлогдсон тэмдэгт мөрөөр эхэлсэн эсэхийг шалгах функц
boolean endsWith(String suffix)	Тодорхойлогдсон тэмдэгт мөрөөр дууссан эсэхийг шалгах функц
int compareTo(String anotherString)	Тэмдэгт мөрүүдийг харьцуулах функц

<code>String replace(char oldChar, char newChar)</code>	Тухайн тэмдэгт мөрийн тэмдэгтийг өөрчлөх
<code>String trim()</code>	Тэмдэгт мөрийн ард болон урд талаас хоосон зайг устгах функц
<code>String toLowerCase()</code>	Тэмдэгт мөрийн бүх үсгийг жижиг болгох функц
<code>String toUpperCase()</code>	Тэмдэгт мөрийн бүх үсгийг том болгох функц

Тэмдэгтийн уртыг олох:

```
int len = s.length();
```

Тэмдэгт мөр дэх тэмдэгтийг авах:

```
String s = "Strings are immutable";
char result = s.charAt(8);
```

Том үсгүүд руу хувиргах, жижиг үсгүүд рүү хувиргах:

```
String Str = new String("Welcome...");
System.out.println(Str.toLowerCase());
System.out.println(Str.toUpperCase());
```

Тэмдэгт мөрийг тасалж авах:

```
String Str = new String("Welcome ...");
System.out.println(Str.substring(10) );
System.out.println(Str.substring(10, 15) );
```

Хэлбэржүүлсэн тэмдэгт мөр үүсгэх:

```
float floatVar = 123.12f;
int intVar = 12;
String stringVar = "Str";
String fs;
fs = String.format("The value of the float variable is " +
    "%f, while the value of the integer " +
    "variable is %d, and the string " +
    "is %s", floatVar, intVar, stringVar);
System.out.println(fs);
```

Тэмдэгт мөрийг гараас унших

Тэмдэгт мөрийг уншихад Scanner классын next(), nextLine() үйлдлүүдийг ашиглана. next() үйлдэл нь үг, үгээр нь унших ба хоосон зайг орхигдуулдаг, nextLine() үйлдэл нь мөрийг бүхэлд нь уншина. Жишээлбэл, хэрэглэгч 'Test reading Strings' гэсэн мөрийг оруулав.

```
String s1 = in.next();
String s2 = in.next()
String s3 = in.nextLine();
System.out.println(s1 + "\n" + s2 + "\n" + s3);
```

Програмын үр дүн дараах байдлаар харагдана.

Test

reading

Strings

Тэмдэгт мөрөөс тэмдэг болон тэмдэгт мөрийн хэсгийг ялгах

Үүний тулд дараах үйлдлүүдийг ашиглаж болно. Үүнд:

```
char charAt(int index)
String substring(int beginIndex)
String substring(int beginIndex, int endIndex)
```

Жишээлбэл, str тэмдэгт мөрөөс 5 индекс дэх тэмдэгтийг устгахдаа substring үйлдлийг ашиглавал дараах хэлбэртэй байна.

```
str = str.substring(0,4) + str.substring(6);
```

Тэмдэгт мөрүүдийг харьцуулах

Жава хэлэнд тэмдэгт мөрүүдийг харьцуулахдаа “==” операторыг ашиглахгүй! Харин үүний оронд equals функцийг ашиглана.

```
if (s1.equals(s2))...
```

Мөн тэмдэгт мөрүүдийг харьцуулахдаа `int compareTo(String anotherString)` функцийг ашиглаж болно. Энэ функц хоёр тэмдэгт мөрийг харьцуулаад дараах тоон утгыг буцаана. Хэрэв функц:

- 0 гэсэн утга буцаавал тэмдэгт мөрүүд тэнцүү байна.
- 0-ээс их эсвэл 0-ээс бага утга буцаавал тэмдэгт мөрүүд тэнцүү биш болохыг илэрхийлнэ.

Жишээ:

```
public class StringComparisonExample
{
    public static void main(String[] args)
    {
        String str = "ABC";
        System.out.println(str.compareTo("DEF"));
        System.out.println(str.compareToIgnoreCase("abc"));
    }
}
```

Програмын үр дүн:

-3

0

Дараах жишээнд тэмдэгт мөрийг палиндром (хоёр талаасаа ижилхэн жишээ нь: abcba) эсэхийг шалгадаг функцийг бичив.

```
private static boolean isPalindromeString(String str)
{
    if (str == null)
        return false;
    int length = str.length();
    for (int i = 0; i < length / 2; i++)
    {
        if (str.charAt(i) != str.charAt(length - i - 1))
            return false;
    }
    return true;
}
```


Тэмдэгт мөрийг тэмдэгт рүү, тэмдэгтийг тэмдэгт мөр рүү хөрвүүлэх жишээ:

```
import java.util.Arrays;

public class StringToCharToString {
    public static void main(String[] args) {
        //String to char array
        String str = "123";
        char[] chArr = str.toCharArray();
        System.out.println("String to char array: "
            +Arrays.toString(chArr));
        //String to char
        char c = str.charAt(1);
        System.out.println("String to char: "+c);
        //char to String
        String s = Character.toString(c);
        System.out.println("char to String: "+s);
        //delete symbols from String
        System.out.println("removing all chars from String: "
            +removeCharFromString("1ABCD12DW",
                '1'));
    }

    private static String removeCharFromString(String str, char c) {
        return str.replaceAll(Character.toString(c), "");
    }
}
```

Программын үр дүн:

```
String to char array: [1, 2, 3]
String to char: 2
char to String: 2
removing all chars from String: ABCD2DW
```

Тэмдэгт мөрийг байт массив руу хөрвүүлэх(мөн эсрэг хөрвүүлэлт) жишээ:

```
import java.util.Arrays;

public class StringByteArray {
    public static void main(String[] args) {
        String str = "www.ulaanbaatar.com";
        // String to byte array
        byte[] byteArr = str.getBytes();
        System.out.println("String to byte array : "
            +Arrays.toString(byteArr));
        // byte array to String
        String str1 = new String(byteArr);
        System.out.println("byte array to String : "+str1);
        //equal ? str and str1
    }
}
```

```

        System.out.println("str == str1? " + (str == str1));
        System.out.println("str.equals(str1)? " +
(str.equals(str1)));
    }
}

```

Программын үр дүн:

```

String to byte array : [119, 119, 119, 46, 106, 111, 117, 114, 110,
97, 108, 100, 101, 118, 46, 99, 111, 109]
byte array to String : www.ulaanbaatar.com
str == str1? false
str.equals(str1)? true

```

StringBuffer класс

String классын объект үүсгэгдсэний дараа өөрчлөгдөх боломжгүй байдаг. Зарим нөхцөлд тэмдэгт мөрийг өөрчлөх шаардлага гардаг. Энэ нөхцөлд StringBuffer классын объектыг хэрэглэнэ.

Тэмдэгт мөрөнд дэд тэмдэгт мөрийг нэмэх, устгах, эсвэл солих буюу тэмдэгт мөрийг өөрчлөх үед StringBuffer классын объект нь өөр дээрээ өөрчлөлтийг хийдэг ба шинээр дахин объект үүсдэггүй.

StringBuffer классын объект үүсгэх

```

StringBuffer str1 = new StringBuffer( );
StringBuffer str2 = new StringBuffer(42);
StringBuffer str3 = new StringBuffer( "New Java lab New world");

```

Огноотой ажиллах

java.time, java.util, java.sql, java.text пакетууд огноо болон цагтай ажиллах классуудыг агуулж байдаг. Жавагийн нийтлэг хэрэглэгддэг огноотой ажилладаг классууд:

```

java.util.Date class
java.sql.Date class
java.util.Calendar class
java.util.GregorianCalendar class
java.util.TimeZone class
java.sql.Time class
java.sql.Timestamp class

```

Жишээ: java.util пакетын Calendar классыг ашиглах

```

Calendar cal = Calendar.getInstance();

//Өнөөдөр 2021.02.20 гэж үзье

```

```
System.out.print("Өнөөдөр " +
    (cal.get(Calendar.MONTH)+1) + "/" +
    cal.get(Calendar.DATE) + "/" +
    cal.get(Calendar.YEAR));
```

Үр дүн:

Өнөөдөр 2/20/2021

Ажил гүйцэтгэх дараалал:

1. Өмнө үүсгэсэн классдаа шаардлагатай Math, String, Date классыг ашиглах функц нэмж бичнэ.
2. Доорх даалгаврууд тус бүрээс 3-г сонгон бичиж ажиллуулах

Суралцахуйн үр дүнг үнэлэх даалгаврууд:

Тэмдэгт мөртэй ажиллах

- a. Үгээс эгшиг үсгүүдийг ялган, тэмдэгт мөр болгож буцаадаг функц бич.
- b. Үгээс гийгүүлэгч үсгүүдийг ялган, тэмдэгт мөр болгож буцаадаг функц бич.
- c. Өгөгдсөн өгүүлбэрээс тоог ялган, тэмдэгт мөр хэлбэрээр буцаадаг функц бич.
- d. Өгөгдсөн үгэнд том үсэг байгаа эсэх шалгах функц бич.
- e. Өгөгдсөн үгэнд жижиг үсэг байгаа эсэх шалгах функц бич.
- f. Өгөгдсөн үгэнд тоо орсон байгаа эсэх шалгах функц бич.
- g. тэмдэгт мөрөөс хоосон зайг олж, таслалаар солих функц бич.
- h. өгөгдсөн тэмдэгт мөрд ... тэмдэгт мөр агуулагдаж байгаа эсэхийг шалгах функц бич.

Математик функцүүдтэй ажиллах

С програмчлалын хэл дээр бодож байсан бодлогуудаас сонгон бодох. Жишээ (А.Хүдэр багийн орчуулсан "Програмчлалын бодлогууд" номоос авав) :

- a. Тэгш өнцөгт гурвалжны катетууд өгөгдсөн бол талбайг ол.
- b. a,b сууриудтай, a-их суурь дахь өнцөг нь α -тай тэнцүү байх адил хажуут трапецийн талбайг ол.
- c. x,y,z бодит тоонууд өгөгдөв. $\max(x + y + z, xyz)$
- d. x,y,z бодит тоонууд өгөгдөв. $\min^2\left(x + y + \frac{z}{2}, xyz\right) + 1$
- e. a,b,c бодит тоонууд өгөгдөв. Хэрэв $a \geq b \geq c$ бол эдгээр тоонуудыг 2 – оор үржүүлж, эсрэг тохиолдолд модулиудаар нь соль.
- f. Хэрэв аль ч хоёр нь тэнцүү биш гурван бодит тоо x,y,z-ийн нийлбэр нь нэгээс бага бол эдгээр гурван тооны хамгийн багынх нь утгыг үлдсэн хоёрынх нь нийлбэрийн хагастай тэнцүү болго. Эсрэг тохиолдолд x,y-ийн багыг нь үлдсэн хоёрын нийлбэрийн хагасаар соль.
- g.
$$f(x) = \begin{cases} x^2 & -2 \leq x \leq 2 \text{ бол,} \\ 4 & \text{эсрэг тохиолдолд;} \end{cases}$$

$$\text{h. } f(x) = \begin{cases} 0 & x \leq 0 \\ x & 0 < x \leq 1 \\ x^2 & \text{бусад тохиолдолд;} \end{cases}$$

i. ГЭХ МЭТ

Ашиглах материал:

- Ю.Намсрай, Т.Гантөр, Д.Ганцоож, Програмчлалын Жава хэл, 2015 он.