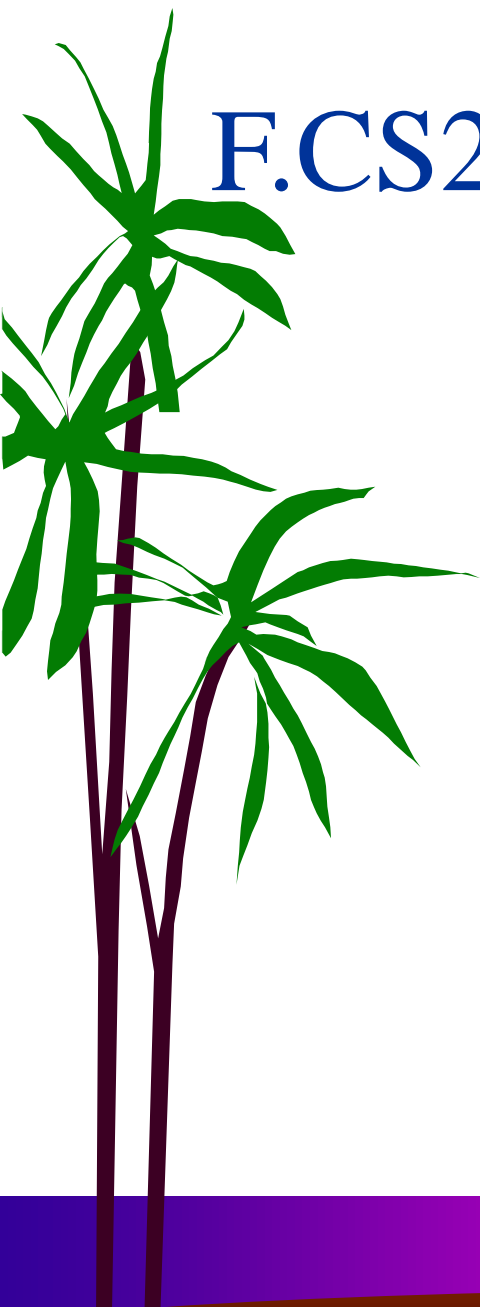
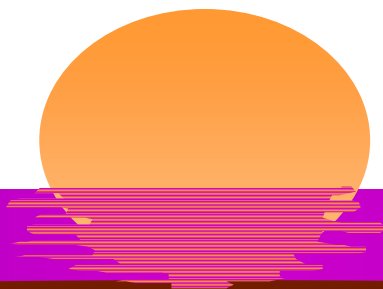


Ү.СS203 - Өгөгдлийн бүтэц ба алгоритм 2022-2023

Д.Золбоо
МТ-н салбарын багш



Хичээлийн зорилго



- Өгөгдлийн бүтэц : дүрслэх, боловсруулах
- Бүх програм өгөгдөлтэй харьцдаг
Тэгэхээр програм өгөгдлийг ямар нэг байдлаар дүрслэх хэрэгтэй.
- Өгөгдөлтэй харьцаж, боловсруулахад алгоритм хэрэгтэй

Хичээлийн зорилго



- Програмыг хөгжүүлэхэд алгоритмын зохиомжийн аргууд хэрэгтэй
- Өгөгдлийн бүтэц, алгоритм бол компьютерийн ухааны суурь

Сурах бичиг



- Sahni Sartaj. Data Structures, Algorithms and Applications in Java. 2000. ISBN 0-07-109217-
- Сургуулийн номын санд 1 хувьтай. Thomas Cormen et al., Introduction to algorithms, MIT press.
- Сурах бичигт орсон жишээний эх код мөн zip форматаар тавигдана.

Тавигдах шаардлага

- Java



- Програмчлалын үндэс



Веб хуудас



- elearn.sict.edu.mn
- Хичээлийн хөтөлбөр, лекцийн үзүүлэн, бие даалт, сурах бичиг, эх код гээд хэрэгтэй бүхнээ олж болно

Бие даалт

- Удирдамжийг удирдлага болгох
- Хугацаандаа өгсөн ажил үнэлэгдэнэ
- Нийт 3 бие даалт





Лаборатори



- Лабораторын ажиллах орчин - Eclipse
(www.eclipse.org)

Үнэлгээ

- 30 - улирлын шалгалт
- 20 - завсрын шалгалт (2 удаа)
- 19 - бие даалт (3 удаа)
- 15 - лабораторын ажил (5 удаагийн лабораторийн ажил)
- 16 – ирц, идэвх (бататгах сорил долоо хоног бүр)

Java



Програмын шинжүүд:

- Зөв үү?
- Програмыг унших, кодыг ойлгоход амархан уу?
- Програм хир баримтжуулагдсан?
- Өөрчлөлт оруулах амархан уу?
- Хир ой шаардагдах вэ?
- Хир удаан ажилах вэ?
- Програм хир нийтлэг вэ?
- Өөрчлөлтгүй өөр машин дээр ажиллах уу?



Програмын бүтэц

Программ класс -> гишүүд (өгөгдөл, функц)

Бие даасан программ -> main()

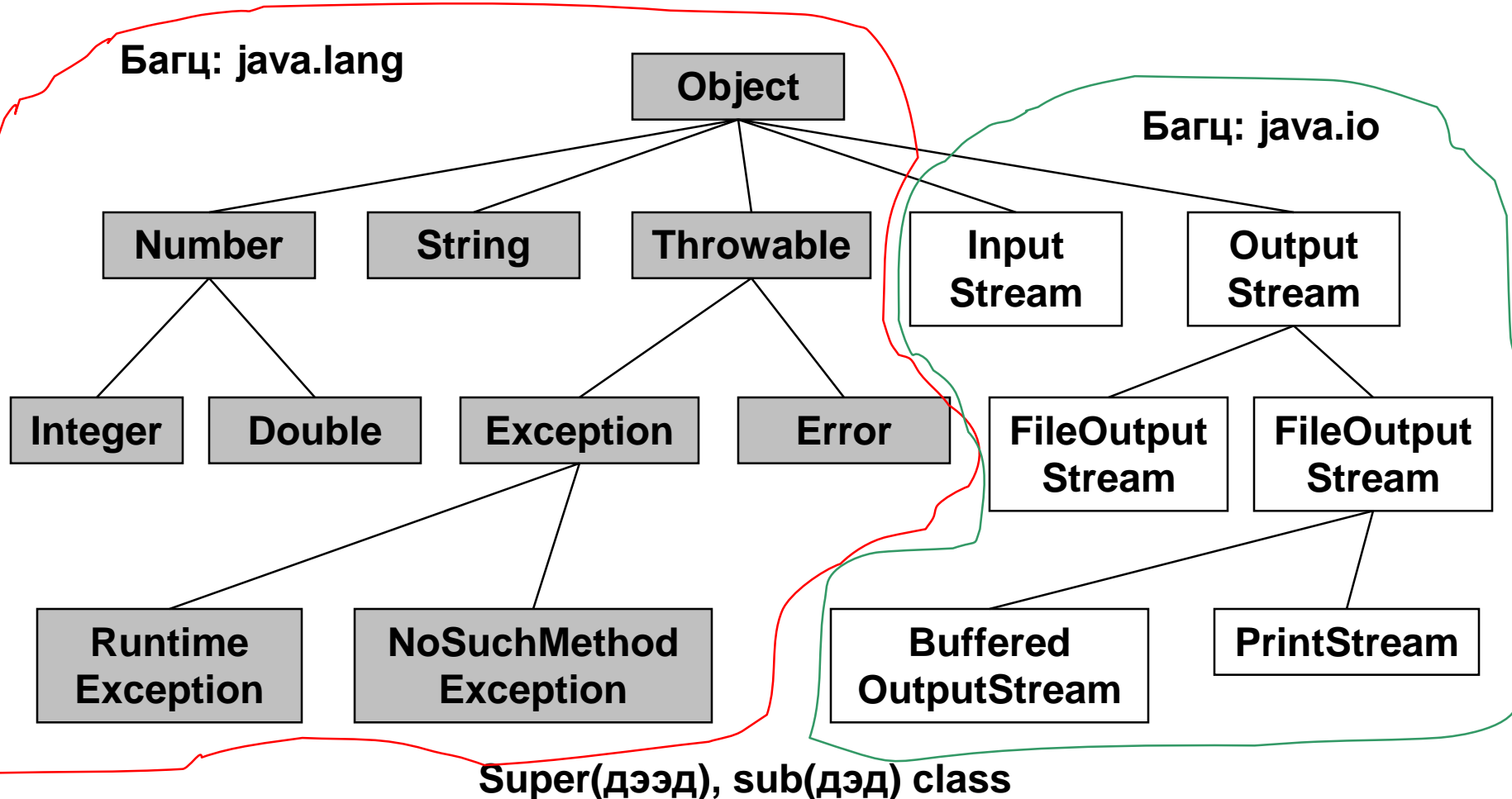
Багц: стандарт багц: java.awt, java.io, java.lang(Integer, String, Boolean ...), java.util

Сурах бичиг: applications, dataStructures, wrappers, exceptions, utilities, misc.

System.out.println("Welcome"); -> System: java.lang багцын класс, out: гишүүн (төрөл нь java.io багцын PrintStream)

Классыг импортлох: import java.io.*

Программын бүтэц



`public class Welcome extends NemaOfSuperClass`

`public final class Welcame`



Java Виртуаль машин, Javadoc

Java хөрвүүлэгч: .java (эх код) -> .class
(байткод)

Java виртуаль машин:

- + машинаас хамаарахгүй, цомхон, найдвартай
- удаан

Javadoc: /** тайлбар */, @param,
@throws, @return ... -> HTML page

Өгөгдлийн төрөл



Төрөл	Утга	Бит	Хүрээ
boolean	False	1	[false, true]
byte	0	8	[-128, 127]
char	\u0000	16	[\u0000, \uFFFF]
double	0.0	64	[+/-4.9e-324, +/-1.8e308]
float	0.0	32	[+/-1.4e-45, +/-3.4e38]
int	0	32	[-2147483648, 2147483647]
long	0	64	+/-9.2e17
short	0	16	[-32768, 32767]

Byte, Integer, Boolean, String -> Объект

Функц



```
public static int abc(int a, int b, int c) {  
    return a+b*c+b/c}
```

- Параметр: зарласан, жинхэнэ
- Утга дамжина -> дотоод хувьсагч шиг
- Давхар ачаалах: `public static float abc(...) {...}`

Онцгой тохиолдол



Дандаа алдаа байдаггүй!

- Унагах: `if(a<0 || b<0 || c<0) throw new IllegalArgumentException ("All parameters must be >=0");`
- Олзолж, боловсруулах: `try {System.out.Println(abc(2,-3,4));} catch (IllefalArgumentException e) { System.out.Println("Positive paraneters are needed!");} finaly { ... }`

Шинэ төрөл



```
public class Currency {  
    public static final boolean PLUS=true, MINUS=false;  
    private boolean sign;  
    private long dollars;  
    private byte cents;  
    public Currency(boolean sign, long dollars, byte cents) { ... }  
    public Currency() { this(PLUS,0L,(byte)0); }  
    public boolean getSign() { ... }  
    public long getDollars() { ... }  
    public void setSign(boolean sign) { ... }  
    public void setDollars(long dollars) { ... }  
    ...  
}
```



Шинэ төрөл

- Өгөгдөл гишүүн (тогтмол): `public static final boolean PLUS=true, MINUS=false;`
- Өгөгдөл гишүүн (тохиолдол): `private boolean sign; private long dollars; private byte cents;`
- Гишүүн функц (байгуулагч): `public Currency(...) { ... } public Currency() { ... }`
- Гишүүн функц: `public boolean getSign() { ... } public long getDollars() { ... } public void setSign(boolean sign) { ... } public void setDollars(long dollars) { ... }`

Хандалтыг өөрчлөгч



Өөрчлөгч	Гишүүний харагдах хүрээ
-	Тухайн багцын классууд
private	Тодорхой C класст
protected	Тухайн багцын классууд, өөр багцын C дэд класст
public	Бүх багцын бүх класст



Удамшил, аргыг орлох(солих)

Шинэ класс дээд классаас удамшдаг:
тохиолдол өгөгдлийн гишүүний хуулбар,
 аргууд удамшдаг.

Currency класс Object классаас удамшсан:
public boolean equals(Object obj); public
String toString(); -> Object класс ->
if(currA.equals(currB) { ... }

Currency.equals() -> Object.equals() –г орлоно



Интерфейс

```
public interface LinkedList {  
    public boolean isEmpty();  
    public int size();  
    public Object get(int index);  
    public int indexOf(Object elem);  
    public Object remove(int index);  
    public void add(int index, Object obj);  
    public String toString();  
}
```

Хаягдал цуглуулах - Garbage Collection



```
int [] a = new int[50000];
```

```
Currency c = new Currency();
```

```
a = null;
```

```
c = null;
```

Хэрвээ a[] int биш Currency байсан бол 7Mb ой хэрэгтэй болно.

Рекурс



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



Өгөгдлийн бүтэц

Өгөгдлийн объект бол тохиолдол/утгын цуглуулга буюу олонлог юм. Жишээ нь:

$\text{integer} = \{0, +1, -1, +2, -2, +3, -3, \dots\}$

$\text{daysOfWeek} = \{S, M, T, W, Th, F, Sa\}$

Өгөгдлийн объект



Тохиолдлууд хамааралтай, хамааралгүй
байж болно

```
myDataObject = { apple, chair, 2, 5.2, red,  
green, Jack }
```

Өгөгдлийн бүтэц



Өгөгдлийн объектын холбоо нь нэг буюу
хэд хэдэн тохиолдол дээр хэрэгжих
үйлдлээр илэрхийлэгддэг

add, subtract, predecessor, multiply

Шугаман (Дараалсан) Жагсаалт

тохиолдлууд дараах хэлбэртэй
байна:

$(e_0, e_1, e_2, \dots, e_{n-1})$

үүнд e_i жагсаалтын элемент

$n \geq 0$ – жагсаалт төгсгөлтэй

түүний хэмжээ нь n

Шугаман жагсаалт

$$L = (e_0, e_1, e_2, e_3, \dots, e_{n-1})$$

Холбоо:

e_0 – 0 дэх (эхний) элемент

e_{n-1} – сүүлийн элемент

e_i - элемент e_{i+1} элементийн өмнө орно

Жишээ/Тохиолдол

CS204 –ийн оюутнууд =
(Бат, Гэрэл, Дулмаа, Оюун, ..., Ганбат)

Days of Week = (S, M, T, W, Th, F, Sa)

Months = (Jan, Feb, Mar, Apr, ..., Nov, Dec)

Шугаман жагсаалтын үйлдэл— size()

Жагсаалтын ХЭМЖЭЭГ ОЛОХ

$$L = (a, b, c, d, e)$$

$$\text{ХЭМЖЭЭ} = 5$$

Шугаман жагсаалтын үйлдэл— `get(theIndex)`

Өгөгдсөн индексийн элементийг авах

$$L = (a, b, c, d, e)$$

$$\text{get}(0) = a$$

$$\text{get}(2) = c$$

$$\text{get}(4) = e$$

$$\text{get}(-1) = \text{error}$$

$$\text{get}(9) = \text{error}$$

Шугаман жагсаалтын үйлдэл — `indexOf(theElement)`

Элементийн индексийг олох

$$L = (a, b, d, b, a)$$

$$\text{indexOf}(d) = 2$$

$$\text{indexOf}(a) = 0$$

$$\text{indexOf}(z) = -1$$

Шугаман жагсаалтын үйлдэл — `remove(theIndex)`

Өгөгдсөн индексийн элементийг
устгаж, буцаах

$$L = (a, b, c, d, e, f, g)$$

remove(2) -с *c* буцах ба

$$L = (a, b, d, e, f, g) \text{ болно}$$

d, e, f, g –ийн индекс 1 –ээр багасна

Шугаман жагсаалтын үйлдэл — `remove(theIndex)`

Өгөгдсөн индексийн элементийг
устгаж, буцаах

$L = (a, b, c, d, e, f, g)$

$remove(-1) \Rightarrow \text{error}$

$remove(20) \Rightarrow \text{error}$