

Національний технічний університет

«Київський політехнічний інститут»

Факультет інформатики і обчислювальної техніки,

Лабораторна робота № 2

|  |
| --- |
| Виконав: |
| студент 3 курсу ФІОТ |
| Групи ІО-91  Матвійчук Б.В. |
|  |

Київ-2011

**package** lab1;

**import** javax.xml.ws.Action;

/\*\*

\*

\* **@author** Богдан Матвійчук

\* группа ІО-91

\* **@param** <T> int\_ob ==> int

\* long\_ob ==> long

\* float\_ob ==> float

\*/

**class** Gen< INT **extends** Integer,LONG **extends** Long, FLOAT **extends** Float>{

INT int\_ob;

LONG long\_ob;

FLOAT float\_ob;

/\*\*

\* **@param** Задаем значения обекта

\*/

Gen(INT a,LONG b,FLOAT c){

int\_ob=a;

long\_ob=b;

float\_ob=c;

}

**void** set\_int (INT a){

int\_ob = a;

}

**void** set\_long (LONG b){

long\_ob=b;

}

**void** set\_float (FLOAT c){

float\_ob=c;

}

/\*\*

**@Action** возвращает значения int

\*/

INT get\_int (){

**return** int\_ob;

}

/\*\*

**@Action** возвращает значения long

\*/

LONG get\_long (){

**return** long\_ob;

}

/\*\*

**@Action** возвращает значения float

\*/

FLOAT get\_float (){

**return** float\_ob;

}

/\*\*

**@Action** выводет на екран обект

\*/

**void** print\_ob (){

System.*out*.print("int = "+ int\_ob + " long = " + long\_ob +

" float = " + float\_ob + '\n');

}

}

**package** lab1;

**class** key\_gen **extends** Gen {

**int** key;

key\_gen (**int** key, Integer a, Long b, Float c) {

**super**(a, b, c);

**this**.key=key;

}

**int** get\_key (){

**return** key;

}

**void** set\_key (**int** key){

**this**.key=key;

}

**void** print\_key (){

System.*out*.print("key = "+ key +" int = "+ int\_ob + " long = " + long\_ob +

" float = " + float\_ob + '\n');

}

}

**package** lab1;

**import** java.util.ArrayList;

**import** java.util.Arrays;

/\*\*

\* **@author** Богдан

\*

\*/

**class** Collection {

**private** **static** **int** *n*;

**private** **static** key\_gen *buf*;

**private** **static** **int** *k*;

/\*\*

\* **@param** n

\* **@return**

\*/

**public** ArrayList<key\_gen> creatColl(**int** n) {

ArrayList<key\_gen> coll = **new** ArrayList<key\_gen>();

**for** (**int** i = 0; i < n; i++) {

*buf* = **new** key\_gen(i, 34 + i, (**long**) 4577777 + i, (**float**) 6 + i);

coll.add(*buf*);

}

**return** coll;

}

/\*\*

\* **@param** coll

\* **@param** n

\* **@param** n1

\* **@return**

\*/

**public** ArrayList<key\_gen> obmen(ArrayList<key\_gen> coll, **int** n, **int** n1) {

*buf* = **new** key\_gen(n, **null**, **null**, **null**);

*buf* = coll.get(n1);

coll.set(n1, coll.get(n)).set\_key(n);

coll.set(n, *buf*).set\_key(n1);

**return** coll;

}

/\*\*

\* **@param** coll

\* **@param** k

\* **@return**

\*/

**public** ArrayList<key\_gen> dell(ArrayList<key\_gen> coll, **int** k) {

**if** (k < coll.size()) {

*buf* = **new** key\_gen(k, **null**, **null**, **null**);

coll.set(k, *buf*);

} **else**

System.*out*.println("Такого поля не існує ! (" + k + ")");

**return** coll;

}

/\*\*

\* **@param** coll

\* **@param** coll\_1

\* **@return**

\*/

**public** ArrayList<key\_gen> addColl(ArrayList<key\_gen> coll,ArrayList<key\_gen> coll\_1) {

System.*out*.println("add\_coll");

*k* = coll.size();

**for** (**int** i = 0; i<coll\_1.size();i++){

coll\_1.get(i).set\_key(*k*+i);

}

coll.addAll(coll\_1);

**return** coll;

}

/\*\*

\* **@param** coll

\* **@param** int\_s

\*/

**public** **void** searchInt(ArrayList<key\_gen> coll, **int** int\_s) {

System.*out*.println("search\_int (" + int\_s + ")");

**for** (**int** i = 0; i < coll.size(); i++) {

**if** (int\_s == coll.get(i).get\_int()) {

coll.get(i).print\_key();

}

}

}

/\*\*

\* **@param** coll

\* **@param** long\_s

\*/

**public** **void** searchLong(ArrayList<key\_gen> coll, **long** long\_s) {

System.*out*.println("search\_long (" + long\_s + ")");

**for** (**int** i = 0; i < coll.size(); i++) {

**if** (long\_s == coll.get(i).get\_long()) {

coll.get(i).print\_key();

}

}

}

/\*\*

\* **@param** coll

\* **@param** float\_s

\*/

**public** **void** searchFloat(ArrayList<key\_gen> coll, **float** float\_s) {

System.*out*.println("search\_float (" + float\_s + ")");

**for** (**int** i = 0; i < coll.size(); i++) {

**if** (float\_s == coll.get(i).get\_float()) {

coll.get(i).print\_key();

}

}

}

/\*\*

\* **@param** coll

\* **@param** float\_s

\*/

**public** **void** searchFloatBinar(ArrayList<key\_gen> coll, **float** float\_s) {

System.*out*.println("search\_float\_Binar (" + float\_s + ")");

sortFloat(coll);

*n*=coll.size();

**int** num = *n*;

**int** index = -1;

**int** indexA = 0;

**while** (num > 0)

{

num = *n* / 2;

**if** (coll.get(indexA + num).get\_float() == float\_s) { index = indexA + num; **break**; }

**if** (coll.get(indexA + num).get\_float() > float\_s) *n* = num;

**if** (coll.get(indexA + num).get\_float() < float\_s)

{

indexA += num;

*n* = *n* - num;

}

}

**if** (index != -1)

System.*out*.println(index);

**else**

System.*out*.println("Значение не найдено");

}

/\*\*

\* **@param** coll

\* **@param** int\_s

\*/

**public** **void** searchIntBinar(ArrayList<key\_gen> coll, **int** int\_s) {

System.*out*.println("search\_float\_Binar (" + int\_s + ")");

sortInt(coll);

*n*=coll.size();

**int** num = *n*;

**int** index = -1;

**int** indexA = 0;

**while** (num > 0)

{

num = *n* / 2;

**if** (coll.get(indexA + num).get\_int() == int\_s) { index = indexA + num; **break**; }

**if** (coll.get(indexA + num).get\_int() > int\_s) *n* = num;

**if** (coll.get(indexA + num).get\_int() < int\_s)

{

indexA += num;

*n* = *n* - num;

}

}

**if** (index != -1)

System.*out*.println(index);

**else**

System.*out*.println("Значение не найдено");

}

/\*\*

\* **@param** coll

\* **@param** long\_s

\*/

**public** **void** searchLongBinar(ArrayList<key\_gen> coll, **long** long\_s) {

System.*out*.println("search\_float\_Binar (" + long\_s + ")");

sortLong(coll);

*n*=coll.size();

**int** num = *n*;

**int** index = -1;

**int** indexA = 0;

**while** (num > 0)

{

num = *n* / 2;

**if** (coll.get(indexA + num).get\_long() == long\_s) { index = indexA + num; **break**; }

**if** (coll.get(indexA + num).get\_long() > long\_s) *n* = num;

**if** (coll.get(indexA + num).get\_long() < long\_s)

{

indexA += num;

*n* = *n* - num;

}

}

**if** (index != -1)

System.*out*.println(index);

**else**

System.*out*.println("Значение не найдено");

}

/\*\*

\* Sort\_int

\* **@param** coll

\* **@return** ArrayList<key\_gen>

\*/

**public** ArrayList<key\_gen> sortInt(ArrayList<key\_gen> coll) {

**int** a,a1;

**for** (**int** i = 0; i < coll.size() - 1; i++) {

**for** (**int** j = 0; j < coll.size()-1; j++){

a=coll.get(j).get\_int();

a1=coll.get(j+1).get\_int();

**if** (a > a1) {

coll= obmen(coll,j, j+1);

}

}

}

**for** (**int** i=0; i<coll.size();i++){

coll.get(i).set\_key(i);

}

**return** coll;

}

/\*\*

\* **@param** coll

\* **@return**

\*/

**public** ArrayList<key\_gen> sortFloat(ArrayList<key\_gen> coll) {

**float** a,a1;

**for** (**int** i = 0; i < coll.size() - 1; i++) {

**for** (**int** j = 0; j < coll.size()-1; j++){

a=coll.get(j).get\_float();

a1=coll.get(j+1).get\_float();

**if** (a > a1) {

coll= obmen(coll,j, j+1);

}

}

}

**for** (**int** i=0; i<coll.size();i++){

coll.get(i).set\_key(i);

}

**return** coll;

}

/\*\*

\* **@param** coll

\* **@return**

\*/

**public** ArrayList<key\_gen> sortLong(ArrayList<key\_gen> coll) {

**long** a,a1;

**for** (**int** i = 0; i < coll.size() - 1; i++) {

**for** (**int** j = 0; j < coll.size()-1; j++){

a=coll.get(j).get\_long();

a1=coll.get(j+1).get\_long();

**if** (a > a1) {

coll= obmen(coll,j, j+1);

}

}

}

**for** (**int** i=0; i<coll.size();i++){

coll.get(i).set\_key(i);

}

**return** coll;

}

/\*\*

\* **@param** coll

\*/

**public** **void** printColl(ArrayList<key\_gen> coll) {

System.*out*.println("print\_coll");

**for** (**int** i = 0; i < coll.size(); i++) {

coll.get(i).print\_key();

}

}

}

**package** lab1;

**import** java.util.ArrayList;

**public** **class** run {

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

INTEGER<Integer> in = **new** INTEGER<Integer> (10);

INTEGER<Integer> in1 = **new** INTEGER<Integer> (12);

INTEGER<Integer> in2 = **new** INTEGER<Integer> (13);

INTEGER<Integer> in3 = **new** INTEGER<Integer> (14);

INTEGER<Integer> in4 = **new** INTEGER<Integer> (15);

System.*out*.println(in1.get\_flag());

System.*out*.println(in2.get\_int(0));

// }

Collection mycoll = **new** Collection ();

ArrayList<key\_gen> c = **new** ArrayList<key\_gen> ();

ArrayList<key\_gen> c1 = **new** ArrayList<key\_gen> ();

c=mycoll.creatColl(10);

c1=mycoll.creatColl(10);

c=mycoll.addColl(c, c1);

c=mycoll.obmen(c,2, 7);

mycoll.searchInt(c,34);

c=mycoll.sortInt(c);

mycoll.printColl(c);

mycoll.searchFloat(c, (**float**) 14.0);

// swing s = new swing (true);

}

}

5

13

add\_coll

search\_int (34)

key = 0 int = 34 long = 4577777 float = 6.0

key = 10 int = 34 long = 4577777 float = 6.0

print\_coll

key = 0 int = 34 long = 4577777 float = 6.0

key = 1 int = 34 long = 4577777 float = 6.0

key = 2 int = 35 long = 4577778 float = 7.0

key = 3 int = 35 long = 4577778 float = 7.0

key = 4 int = 36 long = 4577779 float = 8.0

key = 5 int = 36 long = 4577779 float = 8.0

key = 6 int = 37 long = 4577780 float = 9.0

key = 7 int = 37 long = 4577780 float = 9.0

key = 8 int = 38 long = 4577781 float = 10.0

key = 9 int = 38 long = 4577781 float = 10.0

key = 10 int = 39 long = 4577782 float = 11.0

key = 11 int = 39 long = 4577782 float = 11.0

key = 12 int = 40 long = 4577783 float = 12.0

key = 13 int = 40 long = 4577783 float = 12.0

key = 14 int = 41 long = 4577784 float = 13.0

key = 15 int = 41 long = 4577784 float = 13.0

key = 16 int = 42 long = 4577785 float = 14.0

key = 17 int = 42 long = 4577785 float = 14.0

key = 18 int = 43 long = 4577786 float = 15.0

key = 19 int = 43 long = 4577786 float = 15.0

search\_float\_Binar (14.0)

17