

Exercise 8

Course exam at 24.3. Please register through Oodi.

In the following exercises we implement a *java.util.Set<E>* using closed hashing. Most of the functionality will be inherited from *java.util.AbstractCollection* and *java.util.AbstractSet*. We need to implement basic operations and iterator. Many inherited operations depend on the iterator, but it is needed only if such operations are called. As a hash function we use *hashCode()* method that every *Object* has. We can also assume that all *Objects* have *equals()* method to compare. Each equal object is in the *Set* only once. We do allow *null* element, but do not otherwise limit the element type.

Take a skeleton from course www-page. The skeleton shows which operations you need to implement and which are inherited. Do not use ready collections, just an array of elements as the hash table.

Please study the function of hash table, implementing interfaces, and implementing iterators before starting to plan the exercise. The task is impossible if you do not know how closed hashing works. When you do, it is very easy.

The following task X3 is obligatory for all students. X-tasks must be done **oneself** by each student. Copies/versions of the same answer won't be accepted. Answers must be sent by Wed 15.3. 21:00 using the instructions below. You'll receive an automatic reply by email soon after successful submission. If you won't get the email reply, something went wrong. If the reply contains compiler errors, there is something wrong in the file. Then **resend a fixed version**. The answer must contain a short **self evaluation** where you evaluate the functionality, correctness, time complexity, and possible points of improvement of your solution. A correct self evaluation (for a full answer) is worth one point. The points of these tasks form a part of course evaluation.

Send your solution using a www-form, address and credentials of which you got by email. The solution should be a compilable Java source code file of name *userid.java* where *userid* is the first part of your email address. Also the **class name** of your solution must be *userid*. As the submission is Java source code, the self evaluation must be in comments of the program.

Take a skeleton from course www-page. Do not change the header (name, parameters) of the X-task method(s). Please make sure that the program/class is compilable as such, i.e., have whole answer in the same class and do not use a package.

- X3. Implement *Set* operations *add()* and *contains()* using hash table as described above.
45. Implement *Set* operation *remove()* and update other operations respectively. In *remove*, we replace the removed element with a special *removed* element. Other operations must seek past this element.
46. Implement iteration of the *Set*. In practice, fill up the internal class *HashIter* in the skeleton.
47. Implement rehash of the *Set* to a larger (double size) hash table if we are running out of space.
48. Fill in the course feedback form in Oodi. All comments and suggestions for improvement are welcome, thank you.