Link List: https://docs.oracle.com/javase/8/docs/api/java/util/List.html

Link Arrays: https://www.w3schools.com/java/java\_arrays.asp

GITHUB https://github.com/drtrainig/cap7 (Exemple)

Probleme interesante rezolvate: https://gyansetu-core-java-for-java.gitbook.io/project/untitled-1/creating-and-using-list-set-and-deque-implementations/arraylist

Tema capitolul 7

PROBLEMA 1: Atentie: Pentru problemele 1.a si 1.b este necesar sa creati doar un proiect. (Problema1)

Problema1.a

Sa se creeze o clasa simpla (ProblemaArray) in care se va defini main.

Pentru aceasta problema se cere sa se initieze un array de stringuri citite de la tastatura. (Numarul il stabiliti voi. Alegeti cel putin 3 elemente ca sa nu fie nici 1 sau 2 ca nu ar mai avea sens array-ul).

Se cere sa se construiasca un alt array de stringuri de aceeasi dimensiune in care sa se introduca elementele din primmul array construit in ordine inversa. (Acesta nu il mai veti citi de la tastatura)

Exemplu:

array1 in care elementele sunt citite de la tastatura: mar par cireasa banana

array2 pe care il veti construi pe baza array1: banana cireasa par mar.

Se vor afisa ambele array-uri element cu element folosind parcurgerea cu for.

Problema1.a

Sa se creeze o clasa simpla (ProblemaListe) in care se va defini doar main.

Se cere sa se creeze o lista de numere intregi (folosind List din java).

Folosind o bucla while se cere sa se genereze numere intregi (folosind Random) pana in momentul in care numarul generat este 10. (cand se genereaza 10 se iese din bucla).

Fiecare numar generat pana la intalnirea lui 10 este introdus in lista de mai sus.

Mai departe, daca lista va contine elemente se vor numara cate aparitii ale numarului 5 sunt.

Indicatii:

1.Se va folosi metoda isEmpty() din obietul de tip lista pentru verificare daca lista este goala

Exemplu de rulare:

Presupunem ca s-au generat numerele 2 5 4 16 5 11 18 5 7 10 (la 10 ma opresc)

Introduc numerele intr-0 lista.

Parcurg lista si cand intalnesc valoarea 5 incrementez un contor. (Un contor inseamna ca ne definim o variabila de tip int pe care o incrementam cand intalnim valoarea 5 in lista).

PROBLEMA2

Sa se un program in Java pe baza urmatoarelor cerinte.

1. Sa se creeze o interfata Task in care se va declara metoda executa. ("void executa();")

2. Se va crea o clasa Printer care va implementa implementa interfata Task. Aceasta clasa va avea un parametru de tip String (folositi constructor cu parametri). Metoda execute implementata din interfata va afisa string-ul respectiv.

3. Se va crea o clasa Insumare care va implementa implementa interfata Task. Aceasta clasa va avea doi parametri de tip int(folositi constructor cu parametri). Metoda execute implementata din interfata va afisa suma celor doua numeree.

4. Se va crea o clasa Main cu metoda main unde se va initializa o lista de Task-uri (folosind List din java).

5. Se va crea o bucla for de 10 iteratii in care la fiecare iteratie se va genera un numar random intre 0 si 1. Daca numarul generat este 0 atunci se va introduce in lista un task de tipul Printer. Daca numarul generat este 1 se va introduce in lista un task de tipul Insumare. La constructie alegeti voi parametrii.

6. Se va parcurge lista si se vor executa task-urile fiecarui element.

Probleme optionale:

1. Se da o lista de numere întregi. Sa se afișeze cel mai mare număr din lista

2. Se da o lista de String-uri aleatoriu (construiți voi lista). Se cere sa se afișeze string-ul cu cea mai mare lungime. Dacă sunt mai multe string-uri cu aceeași lungime maxima se vor afișa toate

3. Se dau doua liste de numere întregi (sau string-uri la alegerea voastră). Sa se creeze o lista ce conțin elementele comune din cele doua liste

Link List: https://docs.oracle.com/javase/8/docs/api/java/util/List.html

Link Arrays: https://www.w3schools.com/java/java\_arrays.asp

Exemplu de sortare a unei liste in care folosim o clasa definita de noi: (Optional)

https://howtodoinjava.com/java/sort/collections-sort/

Nota explicativa:

Pentru liste nu putem folosi primitive, doar pentru array-uri asa ca vin in ajutor cu sugestii:

Deci nu pot sa fac. List<int> sau List<double>

In Java exista clasele ajutatoare pentru a ingloba un int intr-o clasa speciala.

Lista de int-uri: List<Integer>

Lista de double: List<Double>

Lista de float-uri: List<Float>

Lista de long-uri: List<Long>

Pentru a adauga elemente in listele de genu, nu este nevoie sa faceti new Integer(2). Se poate face ca si in cazul stringurilor direct prin valoare.

Exemplu la int:

List<Integer> listaDeInt = new ArrayList<>();

listaDeInt.add(22);

listaDeInt.add(65);

listaDeInt.add(21);

Tema capitolul 7

Problema 1.

Sa se creeze un progeram simplu (O clasa Main cu metoda main) in care se va defini un array de elemente de tip int (primitiva). Lungimea array-ului (int) se va citi de la tastatura.

(Folositi <int lungime = Integer.parseInt(scanner.nextLine());> pentru a citi int-ul (contraindicat scanner.nextInt()).

Cu ajutorul unei bucle for se vor introduce in array elemente random intre 0 si 20. Afisati array-ul.

Problema 2.

Sa se creeze un progeram simplu (O clasa Main cu metoda main) in care se va defini un array de elemente de tip String. Lungimea array-ului (int) se va citi de la tastatura.

(Folositi <int lungime = Integer.parseInt(scanner.nextLine());> pentru a citi int-ul (contraindicat scanner.nextInt()).

Cu ajutorul unei bucle for se vor introduce in array elemente citite de la tastatura. Afisati array-ul.

Problema 3.

Sa se creeze un progeram simplu (O clasa Main cu metoda main) in care se va defini o lista de Stringuri (List<String>).

Mai departe se vor citi de la tastatura mai multe String-uri (Alegeti un numar relativ mic sa nu va complicati. de exemplu 3-4 valori sunt de ajuns).

Aceste String-uri se vor introduce in lista doar string-urile citite care nu contin litera "a". (se va verfifica cu metoda contains din String)

Exemplu:

Introduceti cuvantul: birou // O sa il introduc

Introduceti cuvantul: para // Nu sa il introduc

PROBLEMA 4: Atentie: Pentru problemele 4.a si 4.b este necesar sa creati doar un proiect. (Problema1)

Problema4.a

Sa se creeze o clasa simpla (ProblemaArray) in care se va defini main.

Pentru aceasta problema se cere sa se initieze un array de stringuri citite de la tastatura. (Numarul il stabiliti voi. Alegeti cel putin 3 elemente ca sa nu fie nici 1 sau 2 ca nu ar mai avea sens array-ul).

Se cere sa se construiasca un alt array de stringuri de aceeasi dimensiune in care sa se introduca elementele din primmul array construit in ordine inversa. (Acesta nu il mai veti citi de la tastatura)

Exemplu:

array1 in care elementele sunt citite de la tastatura: mar par cireasa banana

array2 pe care il veti construi pe baza array1: banana cireasa par mar.

Se vor afisa ambele array-uri element cu element folosind parcurgerea cu for.

Problema4.b

Sa se creeze o clasa simpla (ProblemaListe) in care se va defini doar main.

Se cere sa se creeze o lista de numere intregi (folosind List din java).

Folosind o bucla while se cere sa se genereze numere intregi (folosind Random) pana in momentul in care numarul generat este 10. (cand se genereaza 10 se iese din bucla).

Fiecare numar generat pana la intalnirea lui 10 este introdus in lista de mai sus.

Mai departe, daca lista va contine elemente se vor numara cate aparitii ale numarului 5 sunt.

Indicatii:

1.Se va folosi metoda isEmpty() din obietul de tip lista pentru verificare daca lista este goala

Exemplu de rulare:

Presupunem ca s-au generat numerele 2 5 4 16 5 11 18 5 7 10 (la 10 ma opresc)

Introduc numerele intr-0 lista.

Parcurg lista si cand intalnesc valoarea 5 incrementez un contor. (Un contor inseamna ca ne definim o variabila de tip int pe care o incrementam cand intalnim valoarea 5 in lista).

PROBLEMA5

Sa se un program in Java pe baza urmatoarelor cerinte.

1. Sa se creeze o interfata Task in care se va declara metoda executa. ("void executa();")

2. Se va crea o clasa Printer care va implementa implementa interfata Task. Aceasta clasa va avea un parametru de tip String (folositi constructor cu parametri). Metoda execute implementata din interfata va afisa string-ul respectiv.

3. Se va crea o clasa Insumare care va implementa implementa interfata Task. Aceasta clasa va avea doi parametri de tip int(folositi constructor cu parametri). Metoda execute implementata din interfata va afisa suma celor doua numeree.

4. Se va crea o clasa Main cu metoda main unde se va initializa o lista de Task-uri (folosind List din java).

5. Se va crea o bucla for de 10 iteratii in care la fiecare iteratie se va genera un numar random intre 0 si 1. Daca numarul generat este 0 atunci se va introduce in lista un task de tipul Printer. Daca numarul generat este 1 se va introduce in lista un task de tipul Insumare. La constructie alegeti voi parametrii.

6. Se va parcurge lista si se vor executa task-urile fiecarui element.

PROBLEMA6

Sa se creeze un program in Java pe baza urmatoarelor cerinte:

1. Se va creea o clasa Produs cu proprietatile denumire si pret. (stabiliti tipurile corect). Constructor cu parametri. Generati metoda equals corespunzator dupa modelul de la curs.

2. Sa se creeze clasa Main cu metoda main in care se va creea o lista de produse. (Presupunem ca este stocul unui magazin)

3. Se da un numar int (n) citit de la tastatura care va determina cate produse se vor citi mai departe (la fel de la tastatura)

4. Mai departe se vor citi n produse de la tastatura. Daca in lista definita la punctul 2 produsul citit este deja, se va afisa un mesaj, daca nu, se va adauga in lista..

5. Afisati lista finala.

Problema suplimentare:

Simulam o clasa de elevi si o catedra de profesori.

Sa se creeze un program prin care se cer urmatoarele cerinte:

//Clase pentru creare

1.1. Se creeaza clasa Curs cu field-urile: nume curs, id curs.

1.2. Se creeaza clasa Persoana cu field-urile: nume complet, varsta.

1.3. Se creeaza clasa Elev care va mosteni clasa Persoana.Clasa Elev va avea ca proprietate in plus o lista de cursuri. (cursurile la care merge elevul)

1.3. Se creeaza clasa Profesor care va mosteni clasa Persoana. Clasa Profesor va avea ca proprietate in plus o lista de cursuri. (cursurile pe care le tine profesorul)

In metoda main se vor creea mai multe obiecte de tip Curs. (Nu este neaparat nevoie sa le cititi de la tastatura. Puteti sa omiteti pentru a va axa pe cerintele principale).

Se va crea un obiect de tipul Student si un Obiect de tipul Profesor. (Incercati sa nu le adaugati aceleasi cursuri la fiecare. Atentie la cerintele principale).

1. Se cere sa se implementeze o metoda/modalitate prin care se determina (se afiseaza) la ce cursuri merge un student, cursuri pe care le tine un profesor.

Cu alte cuvinte vrem sa determinam cursurile comune ale elevului si ale profesorului.

Exemplu:

cursurile elevului <matematica, statistica, programare, filosofie>

cursurile profesorului <matematica, statistica, matematica discreta>

Se va afisa: matematica, statistica.

2. Daca se da un obiect de tip Curs sa se determine daca un student si un profesor au acest curs in comun.