

Programació I

Tema3 - Disseny de composicions iteratives: Esquema de cerca

Universitat de Barcelona
Grau en Enginyeria Informàtica

16 d'octubre de 2012

1 Tema 3. Disseny de composicions iteratives

- Esquemes algorísmics
 - Cerca
- Problemes de composicions iteratives
- Utilitats de classes ja existents

Esquema de cerca

- **Objectiu:** Donada una seqüència d'elements es vol accedir a un element que verifica o compleix una determinada propietat (o condició booleana).
- **Esquema 1:**

```
<inicialitzacions>
elem=Primer();
while ( ! FinalSeq(elem) && !condició_cerca )
{
    elem=Següent(elem);
}
<trobat = !FinalSeq(elem)>
<finalitzacions>
```

Esquema de programació: Cerca

- L'esquema de cerca garanteix que:
 - cada element de la seqüència només es tracta una vegada
 - sempre es tracta l'element actual primer i després s'avança al següent element de la seqüència
 - no es tracta cap element que no sigui de la seqüència
 - a cada iteració, tot element de la part de l'esquerra de la seqüència no compleix la propietat de cerca
 - en acabar, s'ha comprovat que la condició de cerca no es compleix per a tots els elements de la seqüència o bé s'ha trobat el primer element de la seqüència que compleix la propietat de cerca
 - S'ha de garantir que la propietat de cerca es pot avaluar sobre el sentinella de la seqüència.

- Exemple: Donada una seqüència d'enters pel teclat acabada en 0, esbrinar si algun d'aquests enters és la *CLAU_SECRETA* d'accés a un compte.

- 1 Identificació de la seqüència: Seqüència d'enters entrada per teclat:

```
Primer()=sc.nextInt();  
Següent(clau)=sc.nextInt();  
FinalSeq(clau)= (clau == 0)
```

- 2 Identificació de l'esquema: Quan s'entra la clau secreta s'atura el while: esquema de cerca: condició de cerca :
(clau == *CLAU_SECRETA*)

Exemple de resolució II

```
import java.util.Scanner;
public class Cerca {
    public static final int CLAU_SECRETA=725;

    public static void main (String [] args) {
        int clau;    // clau entrada per l'usuari
        Scanner sc;
        sc = new Scanner(System.in);

        System.out.println("Clau_secreta?");
        clau = sc.nextInt();
        while ( (clau != 0) && (clau!=CLAU_SECRETA)) {
            System.out.println("Clau_secreta?");
            clau = sc.nextInt();
        }
        if ( clau!=0 ) {
            System.out.println("Acces_obert");
        } else {
            System.out.println("No_ho_has_encertat!");
        }
    }
}
```

Esquema de cerca

- Un esquema alternatiu que permet no avaluar el sentinella de la seqüència
- **Esquema 2:**

```
<inicialitzacions>
elem = Primer();
trobat = false;;
while ( !FinalSeq(elem) && !trobat) {
    if (condició_cerca) {
        trobat = true;
    } else {
        elem=Següent(elem);
    }
}
<finalitzacions>
```

- Exemple: Donada una seqüència d'enters pel teclat acabada en 0, esbrinar si algun d'aquests enters és la *CLAU_SECRETA* d'accés a un compte.

- 1 Identificació de la seqüència: Seqüència d'enters entrada per teclat:

```
Primer()=sc.nextInt();,  
Següent(clau)=sc.nextInt();,  
FinalSeq(clau)=(clau == 0)
```

- 2 Identificació de l'esquema: Quan s'entra la clau secreta es para:
esquema de cerca: condició de cerca :
(clau == *CLAU_SECRETA*)

Exemple de resolució II

```
import java.util.Scanner;
public class Cerca2 {
    public static final int CLAU_SECRETA=725;
    public static void main (String [] args) {
        int clau;           // clau entrada per l'usuari
        boolean atura;      // indica si s'ha encertat la clau

        Scanner sc;
        sc = new Scanner(System.in);

        System.out.println("Clau_secreta?");
        clau = sc.nextInt();
        atura = false;
        while ( (clau != 0) && !atura) {
            if ( clau == CLAU_SECRETA ) {
                atura = true;
            } else {
                System.out.println("Clau_secreta?");
                clau = sc.nextInt();
            }
        }
    }
}
```

Exemple de resolució III

```
    if ( atura ) {  
        System.out.println("Acces_obert");  
    } else {  
        System.out.println("No_ho_has_encertat!");  
    }  
}  
}
```

Esquema de cerca amb tractament

- Un esquema de cerca on s'ha de realitzar un tractament als elements de la seqüència, a mesura que es comprova la condició de cerca.
- **Esquema 3:**

```
<inicialitzacions>
elem = Primer();
trobat = false;;
while ( !FinalSeq(elem) & & !trobat) {
    <tractament>
    if (condició_cerca) {
        trobat = true;
    } else {
        elem=Següent(elem);
    }
}
<finalitzacions>
```

- Exemple: Donada una seqüència d'enters pel teclat acabada en 0, esbrinar si algun d'aquests enters és la *CLAU_SECRETA* d'accés a un compte i comptar el nombre d'intents que s'han realitzat.

- 1 Identificació de la seqüència: Seqüència d'enters entrada per teclat:

```
Primer()=sc.nextInt();,
```

```
Següent(clau)=sc.nextInt();,
```

```
FinalSeq(clau)=(clau == 0)
```

- 2 Identificació de l'esquema: Quan s'entra la clau secreta es para:
esquema de cerca: condició de cerca :
(clau == *CLAU_SECRETA*)

Exemple de resolució II

```
import java.util.Scanner;
public class Cerca3 {
    public static final int CLAU_SECRETA=725;
    public static void main (String [] args) {
        int      clau;        // clau entrada per l'usuari
        int      intents;     // nombre d'intens de l'usuari
        boolean  atura;       // indica si s'ha encertat la clau
        Scanner  sc;
        sc = new Scanner(System.in);

        intents = 0;
        System.out.println("Clau_secreta?");
        clau = sc.nextInt();
        atura = false;
        while ( (clau != 0) && !atura) {
            intents = intents + 1;
            if ( clau == CLAU_SECRETA ) {
                atura = true;
            } else {
                System.out.println("Clau_secreta?");
                clau = sc.nextInt();
            }
        }
    }
}
```

Exemple de resolució III

```
if ( atura ) {  
    System.out.println("Acces_obert");  
} else {  
    System.out.println("No_ho_has_encertat!");  
}  
System.out.println("Nombre_intents:_ " + intents);  
}  
}
```

Exemple: Donada una frase acabada amb la paraula “fi”, escriu “fi” si totes les paraules son iguals, sino escriu la primera paraula diferent. I

Exemple: Donada una frase acabada amb la paraula “fi”, escriu “fi” si totes les paraules son iguals, sino escriu la primera paraula diferent. I

```
import java.util.*;
public class Iguals {
    public static void main(String[] args) {
        Scanner scan;
        String anterior = "", paraula = "";
        boolean sonIguals = true;

        scan = new Scanner(System.in);
        System.out.println("Introdueix la frase:");
        // Primera parella
        paraula = scan.next();
        if (!paraula.equals("fi")) {
            anterior = paraula;
            paraula = scan.next();
        }
        sonIguals = true;
    }
}
```



Exemple: Donada una frase acabada amb la paraula “fi”, escriu “fi” si totes les paraules son iguals, sino escriu la primera paraula diferent. II

```
while (!paraula.equals("fi") && sonIguals) {  
    if (paraula.equals(anterior)) {  
        // seguent element  
        anterior = paraula;  
        paraula = scan.next();  
    } else {  
        sonIguals = false;  
    }  
}  
if (!sonIguals) {  
    System.out.println(paraula);  
} else {  
    System.out.println("fi");  
}  
}
```



Exemple: Donada una frase acabada amb la paraula “fi”, escriu “si” si hi ha alguna paraula igual a la primera. I

Exemple: Donada una frase acabada amb la paraula “fi”, escriu “si” si hi ha alguna paraula igual a la primera. I

```
import java.util.*;
public class IgualsPrimera {

    public static void main(String[] args) {
        Scanner scan;
        String primera = "";
        String paraula = "";
        boolean sonIguals = true;

        scan = new Scanner(System.in);
        System.out.println("Introdueix la frase:");
        paraula = scan.next();
        if (!paraula.equals("fi")) {
            primera = paraula;
            paraula = scan.next();
        }

        sonIguals = false;
```



Exemple: Donada una frase acabada amb la paraula “fi”, escriu “si” si hi ha alguna paraula igual a la primera. I

```
while (!paraula.equals("fi") && !sonIguals) {  
    if (paraula.equals(primer) ) {  
        sonIguals = true;  
    } else {  
        // següent element  
        paraula = scan.next();  
    }  
}  
if (sonIguals) {  
    System.out.println("si");  
} else {  
    System.out.println("no");  
}  
}
```



Exemple: Donades dues paraules, dir si son iguals o no sense utilitzar el metode equals de la classe String. I

Exemple: Donades dues paraules, dir si son iguals o no sense utilitzar el metode equals de la classe String. I

```
import java.util.*;
public class ParaulesIguals {
    public static void main(String[] args) {
        Scanner scan;
        String primera = "";
        String segona = "";
        boolean sonIguals = true;
        int i;
        int j;

        scan = new Scanner(System.in);
        System.out.println("Introdueix_la_primera_paraula:");
        primera = scan.next();
        System.out.println("Introdueix_la_segona_paraula:");
        segona = scan.next();

        if (primera.length() == segona.length()) {
            i = 0;
```



Exemple: Donades dues paraules, dir si son iguals o no sense utilitzar el metode equals de la classe String. II

```
        while (i<primera.length() && sonIguals) {
            if (primera.charAt(i) == segona.charAt(i)) {
                i++;
            } else {
                sonIguals = false;
            }
        }
    } else {
        sonIguals = false;
    }
    if (sonIguals) {
        System.out.println("si");
    } else {
        System.out.println("no");
    }
}
}
```



Exemple: Donada una frase acabada amb la primera paraula que comença amb “f”, escriu “si”, si hi ha alguna paraula igual a la primera, sense usar el equals de la classe String. I

Exemple: Donada una frase acabada amb la primera paraula que comença amb “f”, escriu “si”, si hi ha alguna paraula igual a la primera, sense usar el equals de la classe String. I

```
import java.util.*;

public class IgualaPrimeraSenseEquals {

    public static void main(String[] args) {
        Scanner scan;
        String primera = "";
        String paraula = "";
        boolean sonIguals = true;
        int i;

        scan = new Scanner(System.in);
        System.out.println("Introdueix la frase:");
```



Exemple: Donada una frase acabada amb la primera paraula que comença amb “f”, escriu “si”, si hi ha alguna paraula igual a la primera, sense usar el equals de la classe String. II

```
paraula = scan.next();
if (paraula.charAt(0)!='f') {
    primera = paraula;
    paraula = scan.next();
}

sonIguals = false;
while (paraula.charAt(0)!='f' && !sonIguals) {

    // Comprovació que es igual a la primera
    if (primera.length() == paraula.length()) {
        sonIguals = true;
        i = 0;
        while (i<primera.length() && sonIguals) {
            if (primera.charAt(i) == paraula.charAt(i)) {
```

Exemple: Donada una frase acabada amb la primera paraula que comença amb “f”, escriu “si”, si hi ha alguna paraula igual a la primera, sense usar el equals de la classe String. III

```
        i++;
    } else {
        sonIguals = false;
    }
}
else {
    sonIguals = false;
}

// pas al següent element en cas que sigui necessari
if (!sonIguals) {
    // següent element
    paraula = scan.next();
}
}
```



Exemple: Donada una frase acabada amb la primera paraula que comença amb “f”, escriu “si”, si hi ha alguna paraula igual a la primera, sense usar el equals de la classe String. IV

```
if (sonIguals) {  
    System.out.println("si");  
} else {  
    System.out.println("no");  
}  
}
```



Problemes I

- ❶ Dissenyar un programar que donat un enter pel teclat, n , compti el nombre de dígit que el componen.
- ❷ Feu un programa que mostri per pantalla els múltiples de 13 positius inferiors a 100.
- ❸ Feu un programa que mostri per pantalla el 100 primers múltiples de 13 positius.
- ❹ Donada una seqüència d'enters pel teclat acabada en 0, feu un programa que esbrini si tots són positius.
- ❺ Donada una seqüència d'enters positius pel teclat acabada en -1, feu un programa que trobi el màxim.
- ❻ Dissenyeu un programa que demani tres nombres a l'usuari, i tot seguit mostri la llista dels múltiples del 3r nombre entrat que hi ha en el rang comprés entre el 1r i el 2n nombre que ha donat l'usuari.
- ❼ Creeu un programa que llegeixi un nombre enter entrat per l'usuari i el descompongui en els seus factors primers.
- ❽ Donada la seqüència dels múltiples de 7 menors de 10000, comptar el nombre de vegades que surt el dígit 2.



Utilitats de la classe String I

- La classe String disposa d'utilitats per a obtenir l'equivalent alfanumèric de diversos tipus de dades.
- Les variables de tipus String disposen d'utilitats (o mètodes) per a treballar amb el valor del seu contingut.

```
public class ClasseString {  
    public static void main (String [] args) {  
        String s1 = String.valueOf(1);  
        String s2 = String.valueOf(2);  
        System.out.println("s1+s2=" + s1 + s2);  
        String s3 = new String("Hola");  
        String s4 = "Hola";  
        System.out.println("s3.length=" + s3.length());  
        System.out.println("s3.majúscules=" + s3.toUpperCase());  
        System.out.println("s3.endsWith(a)=" + s3.endsWith("a"));  
        System.out.println("s3.startsWith(a)=" + s3.startsWith("a"));  
        System.out.println("substring(1,2)=" + s3.substring(1,2));  
        System.out.println("s3.equals_s4=" + s3.equals(s4));  
        System.out.println("s3_==_s4=" + (s3 == s4));  
        System.out.println("Caràcter_a_la_posició_0:" + s3.charAt(0));  
        System.out.println("minúscules_" + s3.toLowerCase());  
    }  
}
```

Utilitats de la classe Character I

- Existeixen unes classes especials per als tipus valor (Integer, Float, Character, Boolean), per a poder proporcionar mètodes de classe bàsics per a operar amb ells.

```
public class ClasseCharacter {  
    public static void main (String [] args) {  
        // Mètodes de classe  
        int codiUNICODE = Character.getNumericValue('A');  
        System.out.println("Unicode(A)="+codiUNICODE);  
  
        boolean digit    = Character.isDigit('A');  
        System.out.println("isDigit(A)="+digit);  
  
        char minuscula = Character.toLowerCase('A');  
        System.out.println("Minuscula(A)="+minuscula);  
    }  
}
```

Utilitats de la classe Character II

```
// Mètodes d'objecte
```

```
Character x = new Character('B');
```

```
Character y = new Character('C');
```

```
boolean equalToC = x.equals(y);
```

```
System.out.println("equals(C)="+equalToC);
```

```
String s = x.toString();
```

```
System.out.println("toString="+s);
```

```
Class c = x.getClass();
```

```
System.out.println("classe="+c.getName());
```

```
}
```

```
}
```