**Esercizi svolti in pseudo-ADA**

Fornire una realizzazione di una **tipo astratto Dizionario** in Ada (si ipotizzi che Chiave e Valore siano DI TIPO Intero e Carattere). Per tale esercizio considerare la specifica sintattica fornita nel seguito

creaDizionario()Dizionario

aggiungi(Dizionario,Chiave,Valore)Dizionario

leggi(Dizionario,Chiave)Valore

cancella(Dizionario,Chiave)Dizionario

uguale(Dizionario,Dizionario)Booleano

Mostrare l'uso del tipo scritto. **Commentare il codice scritto.**

**Package Gennaro\_Dizionario**

**Begin**

Type Dizionario is limited private;

Procedura aggiungi(d: inout Dizionario, c: in Integer, v: in Character);

function leggi(d: in Dizionario, c: in Integer) return Character;

procedure cancella(d:inout Dizionario, c: in Integer);

function uguale( d1: in Dizionario, d2: in Dizionario) return Boolean;

private:

constant max: Positive=200;

type Dizionario is record

chiavi: array [1..max] of Integer;

valori: array[1..max] of Character;

top: Positive=0;

end

**End**

**Package body Gennaro\_Dizionario**

**Begin**

**function cerca(d: in Dizionario, c: in Integer) return Positive**

var i: Positive

begin

if D.top > 0 then

for i=1 to D.top loop

if c= D.chiavi[i] then

return i;

return 0;

end

**procedura aggiungi(d: inout Dizionario, c: in Integer, v: in Character)**

begin

if d.top < max then

if cerca(D,c) =0 begin

D.top=D.top+1;

D.chiavi[D.top]=c;

D.valori[D.top]=v

end

else raise Exception;

end

**function leggi(d: in Dizionario, c: in Integer) return Character**

var x:Positive

begin

x=cerca(D,c);

if x=0 then raise Exception

else return D.valori[x];

end

**procedure cancella(d:inout Dizionario, c: in Integer)**

var x, i:Positive;

begin

x=cerca(D,c);

if x=0 then raise Exception;

else

begin

for i= x to D.top-1 loop

begin

D.chiavi[i]=D.chiavi[i+1]

D.valori[i]=D.valori[i+1];

end

D.top=D.top-1;

end

end

**function uguale( d1: in Dizionario, d2: in Dizionario) return Boolean**

var i:Positive

begin

if d1.top= d2.top then

for i=1 to D1.top loop

begin

x=cerca(d2,d1.chiavi[i])

if x=0 then return false;

else

if d2.valori[x]/=d1.valori[i] then return false;

end

return true;

else return false;

end

**end**

**…**

**With Gennaro\_Dizionario; use Gennaro\_Dizionario;**

**var D1,D2:Dizionario;**

**var x:Positive:**

**inserisci(D1,1,’a’)**

**inserisci(D1,2,’b’)**

**inserisci(D2,1,’c’)**

Fornire una realizzazione di una **classe Dizionario** in Ada (si ipotizzi che Chiave e Valore siano DI TIPO Intero e Carattere). Per tale esercizio considerare la specifica sintattica fornita nel seguito

creaDizionario()Dizionario

aggiungi(Dizionario,Chiave,Valore)Dizionario

leggi(Dizionario,Chiave)Valore

cancella(Dizionario,Chiave)Dizionario

uguale(Dizionario,Dizionario)Booleano

Mostrare l'uso della classe scritta. **Commentare il codice scritto.**

**Generic**

**max:positive**

**Package Dizionario\_Lucia**

**Begin**

**procedure aggiungi( c: in Integer, v:In Character);**

**function leggi(c:in Integer) return Characger;**

**procedure cancella(c: in Integer);**

**end**

**package body Dizionario\_Lucia**

**begin**

**var top: Positive=0;**

**var chiavi: array [1…max] of Integer;**

**var valori: array [1..max] of Character;**

**function cerca(c: in Integer) return Positive**

var i: Positive

begin

if top > 0 then

for i=1 to top loop

if c= chiavi[i] then

return i;

return 0;

end

**procedura aggiungi(c: in Integer, v: in Character)**

begin

if top < max then

if cerca(c) =0 begin

top=top+1;

chiavi[top]=c;

valori[top]=v

end

else raise Exception;

end

**function leggi(c: in Integer) return Character**

var x:Positive

begin

x=cerca(c);

if x=0 then raise Exception

else return valori[x];

end

**procedure cancella(c: in Integer)**

var x, i:Positive;

begin

x=cerca(c);

if x=0 then raise Exception;

else

begin

for i= x to top-1 loop

begin

chiavi[i]=chiavi[i+1]

valori[i]=valori[i+1];

end

top=top-1;

end

end

**end**

**..**

with Dizionario\_Lucia;

package d1 is new Dizionario\_Lucia(3);

package d2 is new Dizionario\_Lucia(300);

with d1;use d1;

with d2, use d2;

d1.aggiungi(1,’a’);

Fornire una realizzazione di una **tipo astratto Dizionario** in Ada (si ipotizzi che Chiave e Valore siano due parametri generici). Per tale esercizio considerare la specifica sintattica fornita nel seguito

creaDizionario()Dizionario

aggiungi(Dizionario,Chiave,Valore)Dizionario

leggi(Dizionario,Chaive)Valore

cancella(Dizionario,Chiave)Dizionario

uguale(Dizionario,Dizionario)Booleano

Mostrare l'uso del tipo scritto. **Commentare il codice scritto.**

Fornire una realizzazione di una **classe Dizionario** in Ada (si ipotizzi che Chiave e Valore siano due parametri generici). Per tale esercizio considerare la specifica sintattica fornita nel seguito

creaDizionario()Dizionario

aggiungi(Dizionario,Chiave,Valore)Dizionario

leggi(Dizionario,Chaive)Valore

cancella(Dizionario,Chiave)Dizionario

uguale(Dizionario,Dizionario)Booleano

Mostrare l'uso della classe scritta. **Commentare il codice scritto.**