

A

1a. feladat:

10

Adjon tesztelő eljárásokat (Axioma_1, Axioma_2) a **verem** alábbi axiómáihoz (3+3):

$$1^{\circ} \text{ Üres?}(\text{Üres}) \wedge \text{VeremMélység}(\text{Üres})=0$$

$$2^{\circ} \text{ Verembe}(v,e)=v' \wedge \neg \text{Üres?}(v') \wedge \text{VeremMélység}(v')=\text{VeremMélység}(v)+1$$

Előbb azonban definiálja az

AllapotKi (Const sElo:String; Const v:TVerem; Const sUto:String)

eljárást ismerve a verem belső ábrázolását, amely legyen láncolt (3).

Felhasználhatja az alábbi, előre definiált konstanst, függvényt és eljárást:

Const e:TElem=... TElem-típusú érték ...

Function CimbolString(Const m:TVeremElemMutato):String

Procedure DeklaracioUtan(Const v:TVerem)

Kérem az ábrázolást is vesse papírra, amely nélkül nem tudom értékelni az AllapotKi eljárást (1)!

2a. feladat:

10

Egy **szimmetrikus** mátrixnak csak az „érdekes” elemeit tároljuk, láncolva.

Típus TMátrix=Rekord(n:Egész, első:TElemCím, hiba:Logikai)

TElemCím=TMátrixElem' Mutató

TMátrixElem=Rekord(ért:TElem, kövOsz, kövSor:TElemCím)

Definiálja az ElemCím függvényt, amely meghatározza egy adott indexű elemnek a memóriabeli helyét. A specifikációja az alábbi:

Függvény ElemCím (Változó m:TMátrix, Konstans s,o:Egész):TElemCím

[Uf: $s \notin [1..m.n] \vee o \notin [1..m.n] \rightarrow m.hiba \wedge \text{ElemCím}(m, s, o) = \text{Sehova} \wedge$

$s \in [1..m.n] \wedge o \in [s..m.n] \rightarrow \text{ElemCím}(m, s, o) = \dots a \text{ tárolt } m_{s,o} \text{ elem címe} \dots \wedge$

$s \in [1..m.n] \wedge o \in [1..s-1] \rightarrow \text{ElemCím}(m, s, o) = \text{ElemCím}(m, o, s)$]

3a. feladat:

10

Írjon egy eljárást, amely egy sor elemei közül **kihagyja** azokat, amelyek egy Tul nevű függvénnyel adott tulajdonságnak eleget tesznek! (7) A kihagyott elemek többé már nem foglalják a helyet a memóriából (1), a sor megmaradt elemei „helyben maradnak”. Mindenek előtt írja le a láncolt ábrázolású sor pontos reprezentációját! (2)

Tervezett ponthatárok:

Kettes, ha legalább:	10	Négyes, ha legalább:	20
Hármas, ha legalább:	15	Ötös, ha legalább:	25
Maximum:		$10+10+10=30$	

B

1b. feladat:

10

Adjon tesztelő eljárásokat (Axioma_1, Axioma_2) a sor alábbi axiómáihoz (3+3):

$$1^{\circ} \text{ Üres?}(\text{Üres}) \wedge \text{SorHossz}(\text{Üres})=0$$

$$2^{\circ} \text{ Sorba}(s,e)=s' \wedge \neg \text{Üres?}(s') \wedge \text{SorHossz}(s')=\text{SorHossz}(s)+1$$

Előbb azonban definiálja az

AllapotKi (Const sElo:String; Const v:TSor; Const sUto:String)

eljárást ismerve a sor belső ábrázolását, amely legyen láncolt (3).

Felhasználhatja az alábbi, előre definiált konstanst, függvényt és eljárást:

Const e:TElem=... TElem-típusú érték ...

Function CimbolString (Const m:TSorElemMutato):String

Procedure DeklaracioUtan (Const v:TSor)

Kérem az ábrázolást is vesse papírra, amely nélkül nem tudom értékelni az AllapotKi eljárást (1)!

2b. feladat:

10

Egy alsóháromszög mátrixnak csak a „nem domináns” elemeit tároljuk, láncolva; a dominánsból csak egyetlen egyet.

Típus TAHMátrix=Rekord (n:Egész, első,felső:TElemCím, hiba:Logikai)

TElemCím=TMátrixElem' Mutató

TMátrixElem=Rekord (ért:TElem, jobbKöv, leKöv:TElemCím)

Definiálja az ElemCím függvényt, amely meghatározza egy adott indexű elemnek a memóriabeli helyét. A specifikációja az alábbi:

Függvény ElemCím (Változó m:TAHMátrix, Konstans s,o:Egész):TElemCím

[Uf: $s \notin [1..m.n] \vee o \notin [1..m.n] \rightarrow m.hiba \wedge \text{ElemCím}(m,s,o)=\text{Sehova} \wedge$
 $s \in [1..m.n] \wedge o \in [1..s] \rightarrow \text{ElemCím}(m,s,o)=\dots \text{a tárolt } m_{s,o} \text{ elem címe} \dots \wedge$
 $s \in [1..m.n] \wedge o \in [s+1..m.n] \rightarrow \text{ElemCím}(m,s,o)=m.felső]$

3b. feladat:

10

Írjon egy eljárást, amely egy verem elemei közül kiiktatja azokat, amelyek egy Tul nevű függvénnyel adott tulajdonságnak nem tesznek eleget! (7) A kiiktatott elemek többé már nem foglalják a helyet a memóriából (1), a verem megmaradt elemei „helyben maradnak”. Mindenek előtt írja le a láncolt ábrázolású verem pontos reprezentációját! (2)

Tervezett pontthatárok:

Kettes, ha legalább:	10	Négyes, ha legalább:	20
Hármas, ha legalább:	15	Ötös, ha legalább:	25
Maximum:		10+10+10=30	

Megoldások

1a. feladat:

Adjon tesztelő eljárásokat (Axioma_1, Axioma_2) a **verem** alábbi axiómáihoz (3+3):

$$1^o \text{ Üres?}(\text{Üres}) \wedge \text{VeremMélység}(\text{Üres})=0$$

$$2^o \text{ Verembe}(v,e)=v' \wedge \neg \text{Üres?}(v') \wedge \text{VeremMélység}(v')=\text{VeremMélység}(v)+1$$

Előbb azonban definiálja az

AllapotKi(**Const** sElo:String; **Const** v:TVerem; **Const** sUto:String)

eljárást ismerve a verem belső ábrázolását, amely legyen láncolt (3).

Felhasználhatja az alábbi, előre definiált konstanst, függvényt és eljárást:

Const e:TElem=... TElem-típusú érték ...

Function CimbolString(**Const** m:TVeremElemMutato):String

Procedure DeklaracioUtan(**Const** v:TVerem)

Kérem az ábrázolást is vesse papírra, amely nélkül nem tudom értékelni az AllapotKi eljárást (1)!

Megoldás:

Type TVeremElemMutato=^TVeremElem;

TVeremElem=Record

érték:TElem;

alatta:TVeremElemMutato;

End;

TVerem=Record

teteje:TVeremElemMutató;

melyseg:Word;

hiba:Boolean;

End;

Procedure AllapotKi(**Const** sElo:String; **Const** v:TVerem;

Const sUto:String);

Begin

Write(sElo); {kísérő szöveg; pl. a változó neve}

Write(' teteje:',CimbolString(v.teteje),

' , mélység:',v.melyseg,' , hiba:',v.hiba);

Write(sUto); {pl.: soremelés}

ReadKey; {nehogy kifusson a képernyőből}

End;

Procedure Axioma_1;

Var db:Word;

l:Boolean;

v:TVerem;

Begin

Writeln('1. axióma -----');

{a deklaráció utáni kezdőállapot:}

DeklaracioUtan(v);

{a deklaráció utáni inicializálás:}

Ures(v);

{az 1. axióma végrehajtása:}

l:=UresE(v); db:=VeremMelyseg(v);

{az 1. axiómában szereplők állapotkiírása:}

Writeln('ÜresE(Ures)=' ,l,' , VeremMelyseg(Ures)=' ,db);

AllapotKi(' Verem=' ,v,CrLf);

End;

```

Procedure Axioma_2;
  Var d1,d2:Word;
        l:Boolean;
        e:TElem;
        v:TVerem;

Begin
  Writeln('2. axióma -----');
  {a deklaráció utáni kezdőállapot;}
  DeklaracioUtan(v);
  {a deklaráció utáni inicializálás;}
  Ures(v);
  {a 2. axióma végrehajtása;}
  d1:=VeremMelyseg(v); Verembe(v,e); l:=UresE(v); d2:=VeremMelyseg(v);
  {a 2. axiómában szereplők állapotkiírása;}
  Writeln('ÜresE(Verembe(v,e))=',l,
        ', VeremMelyseg-változás=',d2-d1);
  AllapotKi(' Verem=',v,CrLF);
End;

```

Az axiómát tesztelő eljárások szempontjából fontos, hogy az

- 1. érintett verem-műveletekkel (Ures, UresE, VeremMelyseg...) kell megfogalmazni az axiómát, és*
- 2. lehessen a belsőállapotot is látni (DeklaracioUtan, AllapotKi).*

Úgyis megfogalmazhatja, hogy egyetlen logikai értéket ír ki az axióma gyanánt. Pl.

```

Procedure Axioma_2;
  Var d1,d2:Word;
        l:Boolean;
        e:TElem;
        v:TVerem;

Begin
  Writeln('2. axióma -----');
  {a deklaráció utáni kezdőállapot;}
  DeklaracioUtan(v);
  {a deklaráció utáni inicializálás;}
  Ures(v);
  {a 2. axióma végrehajtása;}
  d1:=VeremMelyseg(v); Verembe(v,e); l:=UresE(v); d2:=VeremMelyseg(v);
  {a 2. axiómában szereplők állapotkiírása;}
  Writeln('Verembe(v,e)=v'' ÉS NEM ÜresE(Verembe(v,e)) ÉS '
        'VeremMelyseg(v'')=VeremMelyseg(v)+1 :', Not l And d2=d1+1);
  AllapotKi(' Verem=',v,CrLF);
End;

```

1b. feladat:

Adjon tesztelő eljárásokat (Axioma_1, Axioma_2) a sor alábbi axiómáihoz (3+3):

$$1^0 \text{ Üres?}(\text{Üres}) \wedge \text{SorHossz}(\text{Üres})=0$$

$$2^0 \text{ Sorba}(s,e)=s' \wedge \neg \text{Üres?}(s') \wedge \text{SorHossz}(s')=\text{SorHossz}(s)+1$$

Előbb azonban definiálja az

*AllapotKi(Const sElo:String; Const v:TSor; Const sUto:String)
eljárást ismerve a sor belső ábrázolását, amely legyen láncolt (3).*

Felhasználhatja az alábbi, előre definiált konstanst, függvényt és eljárást:

```

Const e:TElem=... TElem-típusú érték ...
Function CimbolString(Const m:TSorElemMutato):String
Procedure DeklaracioUtan(Const v:TSor)

```

Kérem az ábrázolást is vesse papírra, amely nélkül nem tudom értékelni az AllapotKi eljárást (1)!

Megoldás:

Az 1a. megoldás mintájára...

2a. feladat:**10**

Egy szimmetrikus mátrixnak csak az „érdekes” elemeit tároljuk, láncolva.

Típus TMátrix=**Rekord**(n:Egész, első:TElemCím, hiba:Logikai)
TElemCím=TMátrixElem'**Mutató**
TMátrixElem=**Rekord**(ért:TElem, kövOsz, kövSor:TElemCím)

Definiálja az ElemCím függvényt, amely meghatározza egy adott indexű elemnek a memóriabeli helyét. A specifikációja az alábbi:

Függvény ElemCím(**Változó** m:TMátrix, **Konstans** s,o:Egész):TElemCím
[**Uf:** $s \notin [1..m.n] \vee o \notin [1..m.n] \rightarrow m.hiba \wedge ElemCím(m, s, o) = \text{Sehova} \wedge$
 $s \in [1..m.n] \wedge o \in [s..m.n] \rightarrow ElemCím(m, s, o) = \dots a \text{ tárolt } m_{s,o} \text{ elem címe} \dots \wedge$
 $s \in [1..m.n] \wedge o \in [1..s-1] \rightarrow ElemCím(m, s, o) = ElemCím(m, o, s)$]

Megoldás:

Függvény ElemCím(**Változó** m:TMátrix, **Konstans** s,o:Egész):TElemCím
[**Uf:** $s \notin [1..m.n] \vee o \notin [1..m.n] \rightarrow m.hiba \wedge ElemCím(m, s, o) = \text{Sehova} \wedge$
 $s \in [1..m.n] \wedge o \in [s..m.n] \rightarrow ElemCím(m, s, o) = \dots a \text{ tárolt } m_{s,o} \text{ elem címe} \dots \wedge$
 $s \in [1..m.n] \wedge o \in [1..s-1] \rightarrow ElemCím(m, s, o) = ElemCím(m, o, s)$]

Változó hol:TElemCím

ss,oo:Egész

Elágazás

$s \notin [1..m.n]$ vagy $o \notin [1..m.n]$ **esetén**

m.hiba:=Igaz; ElemCím:=Sehova

(2 pont)

$s \in [1..m.n]$ és $o \in [s..m.n]$ **esetén**

hol:=m.elsó

Ciklus oo=2-tól o-ig

hol:=TMátrixElem(hol).kövOsz

Ciklus vége

Ciklus ss=2-tól s-ig

hol:=TMátrixElem(hol).kövSor

Ciklus vége

ElemCím:=hol

(6 pont)**egyéb esetben**

ElemCím:=ElemCím(m, o, s)

(2 pont)**Elágazás vége**

Függvény vége.

2b. feladat:**10**

Egy alsóháromszög mátrixnak csak a „nem domináns” elemeit tároljuk, láncolva; a dominánsból csak egyetlen egyet.

Típus TAHMátrix=**Rekord**(n:Egész, első,felső:TElemCím, hiba:Logikai)
TElemCím=TMátrixElem'**Mutató**
TMátrixElem=**Rekord**(ért:TElem, jobbKöv, leKöv:TElemCím)

Definiálja az ElemCím függvényt, amely meghatározza egy adott indexű elemnek a memóriabeli helyét. A specifikációja az alábbi:

Függvény ElemCím(**Változó** m:TAHMátrix, **Konstans** s,o:Egész):TElemCím
[**Uf:** $s \notin [1..m.n] \vee o \notin [1..m.n] \rightarrow m.hiba \wedge ElemCím(m, s, o) = \text{Sehova} \wedge$
 $s \in [1..m.n] \wedge o \in [1..s] \rightarrow ElemCím(m, s, o) = \dots a \text{ tárolt } m_{s,o} \text{ elem címe} \dots \wedge$
 $s \in [1..m.n] \wedge o \in [s+1..m.n] \rightarrow ElemCím(m, s, o) = m.felső$]

Megoldás:

A 2a. megoldás mintájára...

3a. feladat:**10**

Írjon egy eljárást, amely egy sor elemei közül kihagyja azokat, amelyek egy Tul nevű függvénnyel adott tulajdonságnak eleget tesznek! (7) A kihagyott elemek többé már

nem foglalják a helyet a memóriából (1), a sor megmaradt elemei „helyben maradnak”. Mindenek előtt írja le a láncolt ábrázolású sor pontos reprezentációját! (2)

Megoldás:

Típus TSorElemMutató=TSorElem'Mutató
 TSorElem=**Rekord**(érték:TElem, köv:TSorElemMutató)
 TSor=**Rekord**(eleje,vége:TSorElemMutató, hossz:Egész, hiba:Logikai)

Eljárás Kihagy (Változó s:TSor):
 Változó hol,
 előző:TSorElemMutató
 előző:=Sehova; hol:=s.eleje
Ciklus amíg hol≠Sehova (ciklus-szervezés: 2 pont)
Ha T(TSorElem(hol).érték) **akkor** (Tul-kezelés: 3 pont)
Ha előző=Sehova **akkor** [Tul-k az elején: eleje-t is állítani kell]
 előző:=hol; hol:=TSorElem(hol).köv
 Felszabadít(előző); s.hossz:-1 (felszabadítás:0,5 pont)
 s.eleje:=hol
különben
 TSorElem(előző).köv:=TSorElem(hol).köv
 Felszabadít(hol); s.hossz:-1 (felszabadítás:0,5 pont)
 hol:=TSorElem(előző).köv;
Elágazás vége
különben (nem Tul-kezelés: 1 pont)
 előző:=hol; hol:=TSorElem(hol).köv
Elágazás vége
Ciklus vége
 s.vége:=előző (1 pont)
Eljárás vége.

3b. feladat:

Írjon egy eljárást, amely egy verem elemei közül kiiktatja azokat, amelyek egy Tul nevű függvénnyel adott tulajdonságnak nem tesznek eleget! (7) A kiiktatott elemek többé már nem foglalják a helyet a memóriából (1), a verem megmaradt elemei „helyben maradnak”. Mindenek előtt írja le a láncolt ábrázolású verem pontos reprezentációját! (2)

Megoldás:

A 3a. megoldás mintájára...