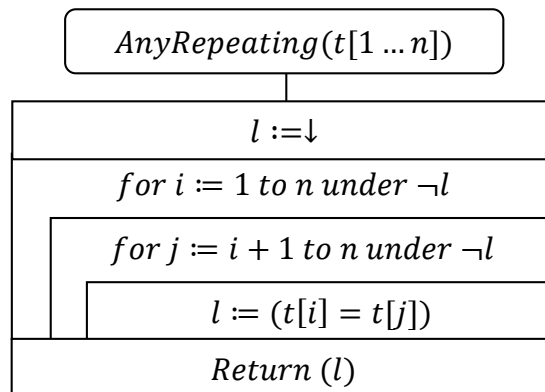


A	Név:	Neptun kód:	Pontszám:
----------	-------------	--------------------	------------------

1. Feladat (10 pont)

Adott a következő algoritmus, amely megállapítja, hogy egy adott számsorozatban van-e ismétlődő érték.



Vezesd le (informálisan) az algoritmus műveletigényét legjobb ($T_{min}(n)$) és legrosszabb ($T_{max}(n)$) esetben, amennyiben minden atomi utasítás (értékadás, kifejezés kiértékelés) 1 műveletigényű, majd add meg nagyságrendileg is az értéket, általános esetben is ($T(n)$). Mi tekinthető legjobb és legrosszabb esetnek egy n hosszú sorozat esetén?

2. Feladat (8 pont)

Szeretnénk takarékos módon szekvenciálisan tárolni a szimmetrikus mátrixokat (ahol az elemek tükörképe szerepel a főátló másik oldalán). Számítsd ki az $n \times n$ -es mátrix helyfoglalását, illetve valósítsd meg az $ind: 1 \dots n \times 1 \dots n \rightarrow \mathbb{N}$ indexfüggvényt, amely megadja az elemek helyzetét.

3. Feladat (12 pont)

Valósítsd meg a verem $Push(s, d)$ és $Top(s)$ műveleteit láncolt reprezentáció mellett. Add meg az algoritmusokat, a műveletigényeket, illetve ábra segítségével magyarázd el a működési elvet.

4. Feladat (12 pont)

Hozd postfix (fordított lengyelform) alakra a következő kifejezést, majd értékeld ki az $u = 100, v = 10, w = 1$ értékekkel. Add meg az átalakítás, illetve a kiértékelés lépéseit oly módon, hogy a bemenet minden elemének feldolgozásakor írd le az aktuális veremtartalmat, valamint a kiírandó értéket (változatlan verem esetén nem szükséges ismételni).

$$x := u - 4 \cdot (w + 4)^2 + (2 - v)$$

5. Feladat (18 pont)

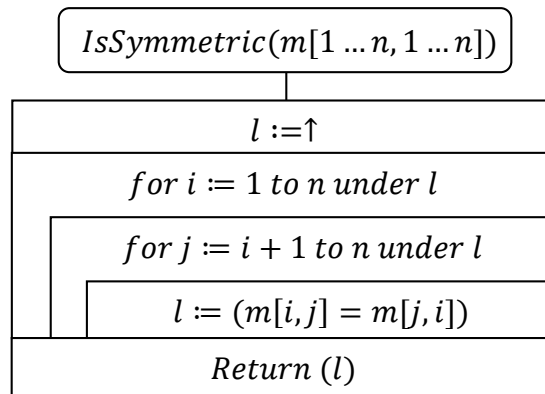
Valósítsd meg a postfix (fordított lengyelforma) alak kiértékelését végző $EvalRPN(in)$ algoritmust, amely visszaadja a kifejezés eredményét. A bemeneten egy (helyes) postfix kifejezés található, amelyben operátorok és operandusok találhatóak (a kifejezésben csak binér műveletek szerepelnek, a változók pedig már behelyettesítésre kerültek). A bemenetet szimbólumonként dolgozzuk fel a $Read(<input>, <symbol>)$ segédművelettel, a bemenet végét $<input> = \varepsilon$ jelzi. Az $IsOperator(<symbol>)$ függvény eldönti, hogy az adott szimbólum operátor-e, az $Eval(<operator>, <leftOperand>, <rightOperand>)$ függvény megadja a művelet eredményét. A megvalósításhoz használj verem adatszerkezetet.

Jó munkát!**Giachetta Roberto**

B	Név:	Neptun kód:	Pontszám:
----------	------	-------------	-----------

1. Feladat (10 pont)

Adott a következő algoritmus, amely megállapítja, hogy egy mátrix szimmetrikus-e.



Vezesd le (informálisan) az algoritmus műveletigényét legjobb ($T_{min}(n)$) és legrosszabb ($T_{max}(n)$) esetben, amennyiben minden atomi utasítás (értékkadás, kifejezés kiértékelés) 1 műveletigényű, majd add meg nagyságrendileg is az értéket, általános esetben is ($T(n)$). Mi tekinthető legjobb és legrosszabb esetnek egy $n \times n$ méretű mátrix esetén?

2. Feladat (8 pont)

Szeretnénk takarékos módon szekvenciálisan tárolni az alsó háromszög mátrixokat (ahol a főátló felett csak 0-k szerepelnek, és a 0-t csak egy példányban tároljuk a memóriába). Számítsd ki az $n \times n$ -es mátrix helyfoglalását, illetve valósítsd meg az $ind: 1 \dots n \times 1 \dots n \rightarrow \mathbb{N}$ indexfüggvényt, amely megadja az elemek helyzetét.

3. Feladat (12 pont)

Valósítsd meg a verem $Pop(s)$ és $Top(s)$ műveleteit aritmetikai reprezentáció mellett. Add meg az algoritmusokat, a műveletigényeket, illetve ábra segítségével magyarázd el a működési elvet.

4. Feladat (12 pont)

Hozd postfix (fordított lengyelforma) alakra a következő kifejezést, majd értékeld ki az $a = 0, b = 2, c = 1$ értékekkel. Add meg az átalakítás, illetve a kiértékelés lépéseit oly módon, hogy a bemenet minden elemének feldolgozásakor írd le az aktuális veremtartalmat, valamint a kiírandó értéket (változatlan verem esetén nem szükséges ismételni).

$$x := (2 - a)^{(3 - c) / 4} + b / 2$$

5. Feladat (18 pont)

Valósítsuk meg a *CorrectParenthesis(in)* algoritmust a helyes zárójelezés azon problémájának eldöntésére, amikor '(', ')', valamint '[,]' zárójelpárok szerepelhetnek a sorozatban, és mindkét zárójel tartalmazhatja a másikat. Pl. '()', '(())', '([])', '[() ([])]' helyes zárójelezések, '(()', '(]', '([)]' nem azok. A megvalósításhoz használj verem adatszerkezetet, a beolvasáshoz a *Read(<bemenet>, <karakter>)* segédfüggvényt. A bemenet végét ϵ jelzi. A bemeneten csak zárójelek szerepelnek.

Jó munkát!

Giachetta Roberto