

Diszkrét modellek alkalmazásai 1. ZH

gépes zárthelyi

2020. 11. 02.

0.1 A zárthelyi szabályai

- A dolgozatra 65 perc kidolgozási idő áll rendelkezésre.
- A feltöltés helye: Cocalc - "HALLGATÓ NEVE - 11_02" mappa.
- A gyakorlatvezető által kiadott segédanyag használata engedélyezett.
- Órai jegyzet, saját jegyzet, más forrásból származó segédanyag használata tilos.
- Kommunikáció csoporttárrsal vagy más egyetemi polgárral (illetve bármilyen személlyel) tilos, akár személyesen, akár online felületen.
- A dolgozattal kapcsolatban a rendelkezésre álló idő alatt szabad kérdezni.
- A dolgozat feladatait .sagews fájlban kell kidolgozni.
- A dolgozat idejére Teams-hívásban kell lenni a gyakorlatvezetővel, amely addig tart, ameddig a dolgozat beadásra nem került. Ezt jelezni kell a gyakorlatvezetőnek, aki leellenőrzi, hogy a beadás megtörtént-e, és így a hallgató csak a megerősítés után hagyhatja el a hívást.
- A fenti szabályok megszegése csalásnak minősül, amely a zárthelyi érvénytelenítését, súlyosabb esetben a gyakorlati jegy megtagadását vonja maga után.

1 A zárthelyi feladatai

1.1 1. feladat 5 pontért:

- 1.1.1 Adott a testresults szótár, amiben hallgatókhoz vannak dolgozat-osztályzatok társítva. Tibi négyest (4) ért el ezen a dolgozaton. István hármast (3), Csenge kettest (2) és Gábornak egyes (1) lett a dolgozat. Hozza létre ezt a szótárat és - beépített függvény segítségével - frissítse Gábor jegyét ötösre (5), valamint adj hozzá Károlyt a listához, aki hármast (3) szerzett! - 3 pont
- 1.1.2 Adj meg a 65445634324 számot prímtényezőik szorzataként - beépített függvény segítségével! - 1 pont
- 1.1.3 Alakítsa szorzattá az $a^2 + 2ab + b^2$ kifejezést! - 1 pont

1.2 2. feladat 6 pontért:

- 1.2.1 Írjon egy függvényt, ami paraméterként kap egy N pozitív egész számot és visszatér az N-edik prímszámmal! - 2 pont
- 1.2.2 Írjon egy függvényt, ami paraméterként kap egy N pozitív egész számot és térjen vissza egy listával, amiben pontosan N darab olyan szám szerepel, amely hárommal vagy öttel osztható, de tizenötös nem osztható! Amennyiben az N szám 3 vagy 5 lenne, akkor térjen vissza egy üres listával! - 4 pont

1.3 3. feladat 7 pontért:

1.3.1 Tekintsük az alábbi gráfot: 1: [0, 1, 2, 5], 5: [2, 3], 2: [3, 0]. Adja meg a gráf egy Hamilton-útját! - 2 pont

1.3.2 Rajzoljuk ki azt az irányított gráfot, amelynek csúcsai $V = 1, 2, 3$; az élei pedig $E = \{(1,1), (1,2), (1,3), (2,2), (2,3), (3,1)\}$! Adja meg az 1-ből a 3-ba menő összes lehetséges utat! - 3 pont

1.3.3 Az alábbi programkódban vagy egy hiba. Hol és mi a hiba? Hogyan működhetne helyesen a feladat - 2 pont

```
#Sage-ben írja ki a képernyőre hogy a (1, 2), (4, 1), (2, 3), (2, 5),  
#(7, 9), (8, 1), (6, 8), (8, 6), (3, 6) #párokkal meghatározott irányított  
#gráfban melyik a maximális hosszúságú út a 4-es csúcsból a 6-os csúcsba,  
#amelyik tartalmazza a 3-as csúcsot!  
G = DiGraph([[1,2,3,4,5,6,7,8,9], [(1, 2), (4, 1), (2, 3), (2, 5), (7, 9),  
(8, 1), (6, 8), (8, 6), (3, 6)]] , loops=True)  
G.show()  
paths = G.all_paths(6, 4)  
paths  
max_path = []  
for p in paths:  
    if 3 in p and len(p) > len(max_path):  
        max_path = p  
max_path
```

1.4 4. feladat 8 pontért:

1.4.1 Írjon függvényt, amely eldönti, hogy a paraméterbeli reláció antiszimmetrikus-e!
- 2 pont

Ha egy reláció antiszimmetrikus, akkor a relációban, ha az (a,b) elem szerepel, akkor csakis akkor szerepel a (b,a) elem, ha a és b egyenlő.

A függvény meghívása: IsAntisimmetric({1, 2, 3, 4}, [[1, 1], [2, 3], [3, 4], [4, 4]]), erre a példára adjon vissza True-t!

1.4.2 Írjon függvényt, amely meghatározza, hogy a paraméterben kapott függvény társít-e minden paraméterben kapott A listabeli elemhez értéket! - 6 pont

A függvény meghívása: IsItCovered([[1,2], [3,5], [5,0]], [1,2,3]), erre a példára adja vissza a (False, [2]) tuple-t, de a IsItCovered([[1,2], [3,5], [5,0]], [1,3])-re adja vissza, hogy (True, []), azaz a tuple első eleme válaszolja meg a kérdést egy logikai igazzal/hamissal, a második eleme pedig legyen egy lista azokról az elemekről, amelyeket nem fed le a függvény!

1.5 5. feladat 9 pontért:

- 1.5.1 Kérjen be a felhasználótól egy N pozitív egész számot! - 1 pont
- 1.5.2 Vizsgálja meg, hogy a bekért N prímszám-e! Ha igen, akkor írassa ki az utána következő 2 prímszám összegét! Ha N nem prímszám, akkor írassa ki az N faktoriális értékét! Mindkét esetben tároljuk el a kiírt értéket egy változóban! - 3 pont
- 1.5.3 Vizsgáljuk meg, hogy az előző feladatban eltárolt érték összetett szám-e (mellőzve a beépített `is_prime()` függvényt)! Írjuk is ki ezt (Igen/Nem)!- 3 pont
- 1.5.4 Végül, - az eredetileg bekért N -hez - konstruálja meg a pontosan $2*N$ hosszú, csupa prímszámokat tartalmazó listát, majd ezt írassa is ki! - 2 pont

2 A zárthelyi sikerességének feltétele: a 35 összpontból 14 pont elérése.