

אלגוריתמים, סמסטר קיץ תשפ"א המכללה האקדמית של ת"א-יפו

פרויקט תכנותי

תאריך הגשה: עד ליום ראשון 15.8.21 בשעה 8:00

הנחיות כלליות

1. הפרויקט ניתן להגשה בבודדים או בזוגות, אך לא בקבוצות גדולות יותר. מומלץ להגיש את הפרויקט בזוגות.
2. איחור בהגשה יאושר רק במקרה של מילואים, מחלה ממושכת או לידה, וגם זאת רק בתנאי שהפנייה למרצה בנושא נעשתה לפני מועד ההגשה המקורי של הפרויקט

מטרת הפרויקט

עליכם לכתוב תכנית בשפת C++ (ובה בלבד. בחירת השפה אינה נתונה לשיקול דעת הסטודנטים), שמוצאת את גרף המסלולים הקצרים ביותר **ממוצא יחיד ליעד יחיד** בגרפים מכוונים, וכן לחשב את זמן הריצה של הפתרון.

השלבים לביצוע הפרויקט

- א. קראו תחילה היטב את ההנחיות של הפרויקט והבינו מה נדרש בדיוק.
- ב. תכננו את ה-design של התוכנית שלכם: בחרו אילו מחלקות תממשו, החליטו על data members מתאימים ועל methods רלוונטיים לכל מחלקה.
- ג. כתבו מימוש מלא לכל המחלקות שייעשה בהן שימוש במסגרת התוכנית.
- ד. ממשו את האלגוריתמים הנדרשים ואת פונקציית ה-main של התוכנית.

תיאור מפורט של מטרת הפרויקט:

קלט: גרף מכוון G ושני קדקודים s ו-t.

פלט: תת-גרף H של G המכיל את כל, ורק את, המסלולים הקצרים ביותר מ-s ל-t. ביתר פרוט: לתת-גרף H תהיה התכונה, שהוא מכיל כל מק"ב מ-s ל-t הקיים ב-G, ויתר על כן, יש בו אך ורק קשתות שהינן חלק ממסלול כזה.

אלגוריתם:

א. הרץ BFS על G מקדקוד s.

- ב. מחק מהגרף כל קשת (u,v) אשר אינה מקיימת את התנאי $d[v] = d[u] + 1$.
 לגרף שמתקבל נקרא G_s .
 ג. בנה את הגרף G_s^T .
 ד. בצע BFS על G_s^T , כאשר קדקוד המוצא הוא t . מחק מהגרף כל קשת (u,v) אשר אינה נגישה מ- t . הגרף שמתקבל הוא H^T .
 ה. מהגרף H^T , בנה את הגרף H .

הנחיות נוספות:

- האלגוריתם צריך להיות מיושם בצורה כזו שהוא מופעל על אובייקט "גרף" המייצג את G ומחזיר אובייקט "גרף" המייצג את H .
- הגרפים אינם חייבים להיות גרפים פשוטים. כלומר יתכנו קשתות עצמיות וקשתות מקבילות.
- האלגוריתם יופעל מתוך תוכנית ראשית המקבלת את G ומדפיסה את H פונקציית `main` של התכנית תקלוט את נתוני הגרף מ-`stdin` בפורמט הבא:
 1.1 מספר שלם, n , שמסמן את מספר הקדקודים בגרף.
 1.2 מספר שלם המייצג את שם קדקוד המוצא s ומספר שלם t המייצג את קדקוד היעד.
 1.3 זוגות של מספרים המייצגים שמות של שני קדקודים i, j (כל קדקוד מיוצג על ידי מספר בין 1 ל- n) שיש ביניהם קשת

דוגמה: אם הקלט הוא

```
5 1 4
1 3 4 2 3 2
3 4 2 1
```

אז הגרף כולל 5 קודקודים שיסומנו ב- $\{1,2,3,4,5\}$, $s=1$, $t=4$, והקשתות הן: $(1,3)$, $(4,2)$, $(3,2)$, $(3,4)$, $(2,1)$.

והפלט יהיה זוגות של קדקודים שהם קשתות הגרף H .

אובייקטים ומבני נתונים בתוכנית:

מבני הנתונים

לצורך ביצוע הפרויקט, הנכם נדרשים לממש את המחלקה הבאה ללא שימוש ב-STL:

גרף מכון שממומש על ידי רשימות שכנויות:

פעולות בסיסיות:

- MakeEmptyGraph(n)** – יצירת גרף ריק מקשתות עם n קדקודים.
- IsAdjacent(u,v)** – מחזיר כן אם הקשת (u,v) שייכת לגרף, ואחרת לא.
- GetAdjList(u)** – החזרת רשימה מקושרת של השכנים של קדקוד u .
- AddEdge(u,v,c)** – הוספת קשת (u,v) בעלת משקל c .
- RemoveEdge(u,v)** – הסרת הקשת (u,v) מהגרף.

constructor אחד או יותר.
destructor שמשחרר את כל הזיכרון שמוחזק ע"י האובייקט.
ReadGraph() פונקציה שתבנה את הגרף מתוך קובץ קלט, ותאפשר את קריאת הקלט במבנה שהוגדר למעלה.
PrintGraph() מדפיסה את הגרף
IsEmpty() מחזירה 1 אם הגרף ריק (אפס קדקודים), 0 אחרת.
AddEdge(int i, int j) מוסיפה צלע (i,j) לגרף. ערך החזרה יהיה 1 אם הצלע נוספה, 0 במקרה תקלה.
Adj(int i) פונקציה שמחזירה את רשימת השכנות של i (הגדירו את האובייקט המתאים שיוחזר מתוך מחשבה על השימושים שהאלגוריתמים עושים בו).

הערות: (1) ייתכן שלא כל ה- Methods יהיו בשימוש בתוכנית זו, אולם עליכם לתכנת גם אותם שכן הם חלק מהפעולות הבסיסיות המוגדרות על גרף.
 (2) הרשימה בהחלט לא מלאה. מותר ורצוי להגדיר Methods נוספים. אפשר גם לעשות וריאציות על ההגדרות שלנו, אם יש לכך סיבה (שכמובן תוסבר בתיעוד שלכם!).
 (3) הגדרת ה- Methods לעצמים נוספים שהגדרתם – באחריותכם!

כמו כן עליכם לממש כמובן כל מחלקה אחרת שלה תזדקקו במהלך התכנית.

בדיקת שגיאות

הקפידו לבדוק שגיאות אפשריות בקלט. במקרה של שגיאה, כתבו הודעת שגיאה למסך `invalid input` וצאו מהתכנית באמצעות הפונקציה `exit(1)`.
 (יש לבצע `#include <stdlib.h>` על מנת להשתמש בה).

מדידת זמני ריצה

עליכם להשתמש בקטע הקוד הבא שמודד את זמן הריצה של הפונקציה `fun` ורושם אותו לקובץ (כמובן תוך ביצוע ההתאמות הדרושות)

```
auto start = chrono::high_resolution_clock::now();

// unsync the I/O of C and C++.
ios_base::sync_with_stdio(false);

fun();// Here you put the name of the function you wish to measure

auto end = chrono::high_resolution_clock::now();

// Calculating total time taken by the program.
double time_taken =
    chrono::duration_cast<chrono::nanoseconds>(end - start).count();

time_taken *= 1e-9;

ofstream myfile("Measure.txt"); // The name of the file

myfile << "Time taken by function <name-of-fun> is : " << fixed
    << time_taken << setprecision(9);
myfile << " sec" << endl;
myfile.close();
```

פלט התוכנית:

1. הדפסה של הגרף המתקבל מהאלגוריתם
2. הדפסה של מדידת זמנים לאלגוריתם

שימו לב להדפסה ברורה וקריאה.

השלבים לביצוע התרגיל

- א. קראו תחילה היטב את ההנחיות של התרגיל והבינו את האלגוריתמים המתוארים בו. מומלץ להריץ את האלגוריתמים "הרצה יבשה" על דוגמאות.
- ב. שימו לב, פתרונות אשר מציעים אלגוריתמים חלופיים לאלה שמתוארים כאן ייפסלו על הסף!
- ג. תכננו את ה-design של התוכנית שלכם: בחרו אילו מחלקות תממשן, החליטו על data members מתאימים ועל methods רלוונטיים לכל מחלקה.
- ד. כתבו מימוש מלא לכל המחלקות שייעשה בהן שימוש במסגרת התוכנית.
- ה. יהיו טסטים במאמא שיכילו בין היתר קלטים שגויים, אם כי התכנית שלכם יכולה בהחלט להיבדק גם על קלטים שלא נמצאים במאמא.

תיעוד:

1. כיתבו בקובץ ה-h. ליד ה- prototype של כל פונקציה מהו הקלט ומהו הפלט שלה.
2. אם יש בפונקציה קוד מסובך שקשה להבינו - הסבירו אותו בהערות בקובץ ה-cpp.
3. בראש קובץ ה-h של כל מודול כיתבו בקצרה מה עושה הקובץ הזה.
4. בראש התוכנית כיתבו בקצרה מה היא עושה, כולל קלט, פלט, מבני נתונים מרכזיים, וכיצד יש להפעיל אותה.
5. וכמובן יש לתעד משתנים, ולהקפיד על שמות משתנים משמעותיים

6. הסבר מפורט של התכנון – מחלקות ועצמים בתוכנית, תפקידיהם, מה השיקולים שעשיתם או הבעיות המיוחדות שפתרתם.

הנחיות הגשה

יש להגיש במערכת mama במקום המיועד להגשה את הקבצים הבאים:
1. קובץ readme שיכיל את כל פרטי ההגשה הבאים:

כותרת – תרגיל תכנות באלגוריתמים תשפ"א סמסטר קיץ.
שורה מתחת - שמות המגישים, מספרי ת.ז. שלהם ומספר הקבוצה של כל אחד מהם (מותר להגיש עם בן זוג מקבוצה אחרת).

שימו לב: קובץ טקסט פשוט – לא word.

2. כל קבצי הקוד בסיומות .cpp ו- .h.

3. רק אחד מבני הזוג יגיש את הפרויקט.

שימו לב! בהמשך יועלה קובץ ובו הנחיות הגשה נוספות

בהצלחה, צוות הקורס ☺