

תרגיל בית מספר 1

נושא: שפת C++ כשפת C משופרת
תאריך הגשה: יום ה', 21/11/2021, 23:59

הגשה ביחידים

בהצלחה רבה!

תיאור התרגיל

מטרתו של תרגיל בית זה הינה כפולה – הראשונה, יישור קו מבחינת רמת התכנות הנדרשת בתחילת הקורס; השנייה, תכנות בשפת C++ כשפת C משופרת, אך לאו דווקא עם דגש על השפה כמונחית עצמים בשלב זה. בתרגיל תממשו מערכת לניקוד משתתפי "משחקי החורף", אשר כוללת מרכיבים סטטיסטיים ומשלבת רכיבי קלט/פלט, עיבוד טקסטואלי, וניהול זיכרון. להנחיות מצורף נספח מתמטי המספק את ההגדרות המתמטיות הנדרשות לפתרון התרגיל.

משחקי החורף 2022: אפיון למערכת ניקוד

גיליונות ניקוד השופטים במשחקי החורף מיוצגים באמצעות וקטורים – והם הבסיס של המערכת לאפיון. יהיה עליכם להגדיר אותם, לצד פעולות הכנסה אל הזיכרון, ושאליות הנוגעות לנתונים שנקלטו עד כה; ההתייחסות למתחרה/שופט/משתמש ממין זכר היא לשם נוחות בלבד. נגדיר **ניקוד שופטים** כווקטור רציף המייצג את ניקוד הביצועים של מתחרה מסוים. **מימד הניקוד** הוא מספר הקואורדינטות המתארות כל ניקוד שופטים – זהו **מספר שלם וידוע מראש הנקבע ע"פ מספר השופטים** במשחקי החורף. נניח כי **כל קואורדינטה מייצגת שופט מסוים**, ע"פ מיפוי מפורש שאינו בהכרח מוצג.

עליכם לתכנן ולממש תכנית לניקוד מתחרים התומכת בפעולות הבסיסיות הבאות:

- [0] יציאה מן המערכת
- [1] הזנת ניקוד שופטים של מתחרה
- [2] הצגת ניקוד שופטים של מתחרה
- [3] הצגת ממוצע ניקוד של מתחרה
- [4] הדפסה של סיכום ממוצע ניקוד המתחרים הקיימים במערכת
- [5] הדפסת וקטור הממוצע של כלל הניקוד (כלומר, ממוצע ניקוד של כל שופט)
- [6] הדפסת מטריצת השונות המשותפת של כלל הניקוד

עם הרצת התכנית, המערכת תציג למשתמש תפריט פעולות התואם את הפעולות הבסיסיות שהוצגו, ותמתין לבחירתו; לאחר ביצוע פעולות [1] עד [6] המערכת תחזור למצב "המתנה" לבחירה הבאה, ורק פעולה [0] תביא לסיומה של התכנית. התקשורת מול המשתמש תתנהל בערוצי הקלט והפלט הסטנדרטיים, **כאשר הודעות שגיאה תצאנה אל ערוץ פלט השגיאה**. במקרה שקלט המשתמש הוא תו פעולה שאינו קיים (תו השונה מהספרות [0-6]), יש להוציא את ההודעה הבאה ולאחר מכן לחזור לתפריט הראשי:

ERROR: invalid command; type 0 for exit

הזנת ניקוד שופטים עבור מתחרה בודד לזיכרון

בבחירת פעולה זו [1], התכנית תקלוט מן המשתמש ניקוד שופטים (ווקטור יחיד) עבור מתחרה יחיד ותשמור אותו במבנה הנתונים הפנימי שלה תחת שם המתחרה שיימסר ע"י המשתמש. קליטת ניקוד השופטים מן המשתמש תתנהל לפי הפרוטוקול הבא:

א. הקלדת שם המתחרה (זהו המזהה הייחודי של כל ניקוד שופטים, המהווה מחרוזת כלשהי בעלת לכל היותר 18 תווים)

והקלדת Enter

ב. הזנת ניקוד השופטים (מספרים ממשיים מטיפוס float) ע"י המשתמש, כאשר כל קואורדינטה תופרד באמצעות פסיק,

והקלדת Enter

ג. עדכון מבנה הנתונים הפנימי של המערכת וחזרה אל התפריט הראשי

תכנות בשפת C++, סתיו 2021-22

במקרה שניקוד עבור שם מתחרה זהה כבר קיים במערכת, עדכון מאוחזר ידרוש לחלוטין כל מידע קודם שהוזן. במקרה שקלט המשתמש חורג מן הפורמט הנדרש, יש להוציא את ההודעה הבאה ולאחר מכן לחזור לתפריט הראשי:

ERROR: invalid input

ראו התייחסות בהמשך לתסריט בו יוזנו וקטורי ניקוד מעבר לקיבולת הזיכרון של המערכת.

הצגת ניקוד שופטים של מתחרה

בבחירת פעולה זו [2], על המערכת להדפיס לערוץ הפלט הסטנדרטי ניקוד שופטים של מתחרה נתון ע"פ שמו. הפרוטוקול כולל את הקלדת שם המתחרה ולאחר מכן Enter. בהינתן שם חוקי אשר קיים בזיכרון, יודפס השם וניקוד השופטים בפורמט וקטור שיתואר בהמשך.

בהינתן שם שאינו קיים בזיכרון (מחרוזת המיוצגת ע"י `<name>`), יש להוציא את ההודעה הבאה ולאחר מכן לחזור לתפריט הראשי:

ERROR: `<name>` does not exist

הצגת ממוצע ניקוד של מתחרה

בבחירת פעולה זו [3], על המערכת להדפיס לערוץ הפלט הסטנדרטי את ממוצע הניקוד של מתחרה נתון ע"פ שמו. הפרוטוקול כולל את הקלדת שם המתחרה ולאחר מכן Enter. בהינתן שם חוקי אשר קיים בזיכרון, יודפס השם וממוצע הניקוד (עם שתי ספרות אחרי הנקודה לכל היותר); אין להדפיס נקודה עשרונית אם אין צורך בכך), מופרדים באמצעות פסיק ורווח. בהינתן שם `<name>` שאינו קיים בזיכרון, יש להוציא את ההודעה הבאה ולאחר מכן לחזור לתפריט הראשי:

ERROR: `<name>` does not exist

הדפסת סיכום הניקוד

בבחירת פעולה זו [4], המערכת תדפיס לערוץ הפלט הסטנדרטי רשימה הכוללת את כל המתחרים עבורם הוזן ניקוד עד כה למערכת. כל שורה תכיל את שם המתחרה ואת הניקוד הממוצע שלו (עם שתי ספרות אחרי הנקודה לכל היותר); אין להדפיס נקודה עשרונית אם אין צורך בכך), מופרדים באמצעות פסיק ורווח. סדר המתחרים הינו סדר ההכנסה המקורי שלהם למערכת; לדוגמא:

CharlesDickens, 8.56

OscarWilde, 8.74

PapaDimitriou, 8.02

VictorHugo, 8.67

במקרה שזיכרון המערכת ריק בשלב של בחירת פעולה זו, יש להוציא את ההודעה הבאה ולאחר מכן לחזור לתפריט הראשי:

ERROR: no records in the system

הדפסת ווקטור הממוצע של הניקוד הקיים בזיכרון

בבחירת פעולה זו [5], התכנית תחשב את ווקטור הממוצע של הניקוד הקיים בזיכרון ותדפיס אותו אל ערוץ הפלט הסטנדרטי. ווקטור הממוצע מוגדר כווקטור רציף במימד הניקוד אשר כל קואורדינטה שלו מהווה את ממוצע הניקוד של אותו שופט (ראו פירוט בנספח). ההדפסה תתבצע בפורמט וקטור כפי שיתואר בהמשך. במקרה שזיכרון המערכת ריק בשלב של בחירת פעולה זו, יש להוציא את ההודעה הבאה ולאחר מכן לחזור לתפריט הראשי:

ERROR: no records in the system

בכל מקרה אחר, המערכת תחזור לתפריט הראשי לאחר החישוב וההדפסה.

הדפסת מטריצת השונות של הניקוד הקיים בזיכרון

בבחירת פעולות זו [6], התכנית תחשב את מטריצת השונות המשותפת (*covariance matrix*) של כלל הניקוד הקיים בזיכרון ותדפיס אותה אל ערוץ הפלט הסטנדרטי. מטריצת השונות המשותפת של וקטורי הניקוד הינה מטריצה ריבועית במימד הניקוד אשר כל אלמנט שלה מכמת מדד קשר בין קואורדינטות מסוימות עבור כלל הניקוד. הסבר מפורט ומתכון לחישוב המטריצה מופיעים בנספח. במקרה שזיכרון המערכת ריק בשלב של בחירת פעולה זו, יש להוציא את ההודעה הבאה ולאחר מכן לחזור לתפריט הראשי:

ERROR: no records in the system

בכל מקרה אחר, המערכת תחזור לתפריט הראשי לאחר החישוב וההדפסה.

תכנות בשפת ++C, סתיו 2021-22

פורמט הדפסת וקטור: ההדפסה תכיל את שם המתחרה (או במקרה של חישוב וקטור הממוצע את המחרוזת mean), סימן שווה, וסוגר מרובע שמאלי – ולאחר מכן את כל הקואורדינטות של הווקטור – מופרדות באמצעות פסיק – ולסיום, סוגר מרובע ימני. לדוגמא, כך תיראה הדפסה של וקטור הממוצע (פעולה [5]) עבור וקטורי ניקוד ממימד 6:

```
mean=[7.81,8.51,6.66,7.1,8.82,9.01]
```

פורמט הדפסת מטריצת השונויות המשותפת: השורה הראשונה תכיל את המחרוזת cov, סימן שווה, וסוגר מרובע שמאלי – מופרדים באמצעות רווח. כל אחת מן השורות הבאות תכיל שורת אלמנטים של המטריצה, כאשר כל אלמנט בשורה יופרד מאחר באמצעות פסיק בודד. השורה האחרונה בהדפסה, שתבוא לאחר השורה האחרונה של המטריצה, תכיל רק סוגר מרובע ימני. לדוגמא, מטריצת שונויות משותפת ממימד 4 תודפס כך לפלט הסטנדרטי:

```
cov=[
1,0.5,0,2.55
0.5,1,0,0.7
0,0,1,0
2.55,0.7,0,1
]
```

הרצת התכנית

בעת הרצת התכנית יימסרו לה שני ארגומנטים: הראשון יציין את מימד הניקוד (כלומר, מספר השופטים) והשני יציין את מספר המתחרים (וקטורי הניקוד) המקסימלי שעל המערכת להכיל. לדוגמא, לאחר קומפילציה של התכנית לכדי קובץ הרצה בשם CanadaWinterGames2022, ההרצה הבאה בטרמינל תאתחל מערכת המיועדת לקליטת וקטורי ניקוד ממימד 16 ולכל היותר 100 במספר:

```
>> ./CanadaWinterGames2022 16 100
```

כל ניסיון להריץ את התכנית אחרת, או הרצה עם פרמטרים שגויים, תביא להודעת השגיאה הבאה:

```
ERROR: USAGE <executable> NUM_REFEREEES MAX_ATHLETES
```

הנחות עבודה ופורמט

- על מבנה הנתונים בו מאוחסנים וקטורי הניקוד להיות מוקצה דינאמית על ה-Heap. יש לכם חופש מוחלט בבחירת טיפוס מבנה הנתונים.
- שם מתחרה הינו מחרוזת אלפא-נומירית ייחודית שאינה עולה באורכה על 18 תווים כלשהם; השמות cov, mean הינם שמות שמורים בהם לא ניתן לעשות שימוש.
- יש להדפיס ערכים עם שתי ספרות אחרי הנקודה לכל היותר; אין להדפיס נקודה עשרונית אם אין צורך בכך.
- במקרה של דריסת נתונים במהלך ריצת התכנית (עדכון של ניקוד עבור מתחרה מסוים) – ההדפסה תהיה לפי סדר ההכנסה המקורי.
- מספר וקטורי הניקוד שניתן לאחסן בזיכרון חסום ע"י הארגומנט השני שניתן לתכנית בעת הרצתה; ניסיון להזין וקטורי ניקוד באמצעות פעולה [1] מעבר לחסם זה, יגרור הודעת שגיאה המציינת כי מספר המתחרים המקסימאלי כבר מאוחסן בזיכרון:

```
ERROR: MAX_ATHLETES exceeded
```

תכנות בשפת C++, סתיו 2021-22

דגשים

- יש לתכנן מראש את מבנה התכנית, ולהגדיר בהתאם את המרכיבים איתם תעבדו.
 - יש לממש את עבודת הקלט והפלט באמצעות רכיבי C++ ולא באמצעות פונקציות C.
 - הקצאה דינאמית צריכה להתבצע רק באמצעות הפקודות new ו-delete.
 - עבודה עם מחרוזות יכולה להתבצע באמצעות טיפוס `std::string`, אך אין דרישה לכך.
 - בתרגיל בית זה אין להשתמש בספריית STL; מבני הנתונים והאלגוריתמים צריכים להיות ממומשים על-ידיכם.**
 - יש לבדוק תקינות קלטים ולהציג הודעות שגיאה מתאימות.
 - עליכם לוודא כי התכנית עוברת קומפילציית g++ התואמת את הקומפיילר שעל שרת החוג ללא כל שגיאות או אזהרות כלשהן, ורצה בהצלחה.
 - עליכם לתעד את הקוד באמצעות הערות המתארות בקצרה את המודולים והפונקציות השונים.
 - יש להריץ את הבודק האוטומטי על שרת החוג בטרם ההגשה בכדי לוודא תאימות ונכונות של ההגשה:
- התחברו לשרת החוג והריצו **hwcheck** על הארכיב שלכם, או לחילופין העלו את הארכיב תוך שימוש בפרוטוקול HTML בקישור <http://cs.telhai.ac.il/homework>

הגשה

- עליכם להגיש במערכת Moodle קובץ ארכיב מטיפוס zip בלבד, ששמו כולל את קוד הקורס ('31'), שם התרגיל ('ex1') ותעודת הזהות של הסטודנט/ית המגיש/ה, מופרדים בקו תחתי בפורמט הבא: **31_ex1_studID.zip**.
- על ארכיב zip זה להכיל את כל קבצי המקור (ממשק/מימוש) הנדרשים לקומפילציה, והוא רשאי להכיל תיעוד טקסטואלי; מבחינת טיפוס קבצים, עליו לכלול רק קבצים עם סיומות ***.txt *.h *.cpp**.
- לדוגמה: על סטודנט שמספר הזיהוי שלו הינו 012345678 להגיש ארכיב בשם **31_ex1_012345678.zip** הכולל את כל קבצי המקור של הפרוייקט, ללא תיקיות כלשהן, ורשאי להכיל קובץ טקסטואלי לתיעוד.

אי-הקפדה על ההנחיות, כולל פורמט ההגשה הדיגיטלי, תגרור הורדה בציון התרגיל.
לא תתקבלנה הגשות באיחור!

בהינתן M ווקטורים ממימד n , $\vec{x}^{(1)}, \vec{x}^{(2)}, \dots, \vec{x}^{(M)} \in \mathbb{R}^n$, התוחלת של קבוצת הווקטורים הללו היא ווקטור ממשי ממימד n בעל הצורה המפורשת הבאה:

$$\vec{\mu} = \begin{pmatrix} \frac{1}{M} \cdot \sum_{i=1}^M x_1^{(i)} \\ \frac{1}{M} \cdot \sum_{i=1}^M x_2^{(i)} \\ \vdots \\ \frac{1}{M} \cdot \sum_{i=1}^M x_n^{(i)} \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{pmatrix}$$

כלומר, ווקטור בו כל קואורדינטה מהווה את הממוצע של הקואורדינטות המקבילות עבור כל M הווקטורים.

מטריצת השונות המשותפת הינה מטריצה ריבועית מעל הממשיים ממימד n אשר אלמנט ij שלה מתייחס לקשר בין קואורדינטה i לבין קואורדינטה j עבור כלל הווקטורים:

$$\mathbf{C} = (c_{ij}) :$$

$$\begin{cases} c_{ij} = \frac{1}{M} \cdot \sum_{k=1}^M (x_i^{(k)} - \mu_i) \cdot (x_j^{(k)} - \mu_j) & M = 1 \\ c_{ij} = \frac{1}{M-1} \cdot \sum_{k=1}^M (x_i^{(k)} - \mu_i) \cdot (x_j^{(k)} - \mu_j) & M > 1 \end{cases}$$

שימו לב שהנירמול משתנה בין מקרה בו יש ווקטור בודד ($M=1$) לבין מקרה של ווקטורים מרובים.