

תרגיל בית: מהדורה מיוחדת

נושא: אמולטור קבצים

דדליין: יום ג', 15/02/2022, 23:59

הגשה ביחידים

בהצלחה רבה!

תיאור התרגיל

תרגיל זה עוסק במערכת הקבצים ובו תשדרגו את טיפוס הקובץ הסטנדרטי הקיים בשפה כברירת מחדל. מטרת התרגיל היא תכנון ומימוש של אובייקט עטיפה המשדרג את הקובץ הסטנדרטי, וכן תרגול תכנותי של פקודות קבצים הנתמכות ע"י מנגנון חריגות מובנה.

חלק א': מימוש אובייקט קובץ מוכלל

בחלק זה של התרגיל עליכם לממש אובייקט קובץ מוכלל, באמצעות עטיפת טיפוס ברירת המחדל והוספת פונקציונליות, תחת דרישה מפורשת למימוש של המרכיבים הבאים:

- [1] אופרטור [] לקריאה בלבד של תו בודד בגישה אקראית; פונקציה זו תחזיר את התו הכתוב בקובץ נתון ע"פ מיקומו. באם המיקום חורג מגבולות הקובץ הקיים, תיזרק חריגה.
- [2] אופרטור [] לכתיבה של תו בודד בגישה אקראית; פונקציה זו תכתוב תו נתון בקובץ במיקום נתון. באם המיקום חורג מגבולות הקובץ הקיים (החל מאינדקס אחד לאחר סוף הקובץ), תיזרק חריגה.
- [3] פונקציית "נגיעה בקובץ" בשם touch; פונקציה זו תעדכן את חתימת הזמן של קובץ במערכת הקבצים ע"פ שם שיינתן לה כארגומנט, או תיצור קובץ ריק בשם זה, באם לא היה קיים. בכל מקרה, פונקציה זו אינה משנה את תוכנו של הקובץ.
- [4] פונקציית העתקת קובץ בשם copy; פונקציה זו תעתיק את תוכנו של קובץ מקור לקובץ מטרה, ע"פ שמותיהם הנתונים. קובץ המטרה אינו בהכרח קיים לפני הרצת הפקודה.
- [5] פונקציית מחיקת קובץ בשם remove; פונקציה זו תמחוק את הקובץ הנתון ממערכת הקבצים.
- [6] פונקציית העברת קובץ בשם move; פונקציה זו תעתיק את תוכנו של קובץ מקור לקובץ מטרה, ע"פ שמותיהם הנתונים, ולאחר מכן תמחוק את קובץ המקור. קובץ המטרה אינו בהכרח קיים לפני הרצת הפקודה.
- [7] פונקציית הדפסה בשם cat; פונקציה זו תדפיס את כל תוכנו של הקובץ.
- [8] פונקציית הדפסת רישא בשם head; פונקציה זו תדפיס את 10 השורות הראשונות של הקובץ כאשר תיקרא ללא פרמטרים, או תדפיס את n השורות הראשונות של הקובץ בהינתן n כארגומנט אופציונלי.
- [9] פונקציית הדפסת סיפא בשם tail; פונקציה זו תדפיס את 10 השורות האחרונות של הקובץ כאשר תיקרא ללא ארגומנטים, או תדפיס את n השורות האחרונות של הקובץ בהינתן n כארגומנט אופציונלי.

מעבר לדרישות אלו, אתם חופשיים במימוש טיפוס זה כל עוד הפונקציונליות של הקובץ נשמרת. מספר דגשים נוספים:

- רשימת פעולות הקבצים [1-9] לא בהכרח תמופה לקובץ הכותר של אובייקט הקובץ; כך למשל, אובייקט המשודרג יפורק ע"י ישות אחרת (מערכת הקבצים), ולא ע"י עצמו, וכך גם לגבי יצירתו באמצעות touch.
- כל שגיאה הנובעת משימוש בקבצים (למשל, אי-קיום של קובץ המושא לפעולה), בנוסף לשגיאות המפורשות של האופרטורים (1)+(2), צריכה להיות מטופלת באמצעות מנגנון חריגות מובנה שיתוכנן וימומש על-ידכם.

תכנות בשפת ++C, סתיו 2021-22

חלק ב': אמולטור קבצים

בחלק זה של התרגיל עליכם לממש תכנית המדמה סביבת טרמינל המוגבלת למספר פקודות קבצים, בהתבסס על טיפוס הקובץ המוכלל שמימשתם בחלק א'.

הרעיון הוא שבהרצת התכנית שלכם, ייפתח ערוץ קלט להזנת פקודות קבצים מצד המשתמש, **בהסתמך על הפקודות [1-9]** הנ"ל. התחביר החוקי המתאים לפקודות [1-9] בתכנית שלכם מתואר באמצעות המפרט הבא, הכולל גם פקודת יציאה [10]:

```
[1]      read FILENAME POSITION
[2]      write FILENAME POSITION CHARACTER
[3]      touch FILENAME
[4]      copy SOURCE_FILENAME TARGET_FILENAME
[5]      remove FILENAME
[6]      move SOURCE_FILENAME TARGET_FILENAME
[7]      cat FILENAME
[8]      head FILENAME [n]
[9]      tail FILENAME [n]
[10]     exit
```

הקלדת כל אחת מן הפקודות צריכה לכלול את שם הפקודה והארגומנטים הרלוונטיים, מופרדים ברווח, ולסיום ENTER. שימו לב שפקודות [1, 7-9] הן בעלות פלט, אשר צריך להיות מודפס לערוץ הפלט הסטנדרטי מיד לאחר הרצתן; כל שאר הפקודות נטולות פלט, והתכנית אמורה לחזור למצב האזנה לקלט מן המשתמש מיד לאחר ביצוען.

הודעות שגיאה

במקרה של חריגה מן הפרוטוקול המתואר, או במקרה של שגיאת קבצים / זמן-ריצה, על התכנית להדפיס הודעת שגיאה מתאימה ולחזור למצב האזנה לקלט מן המשתמש; על הודעות השגיאה בתרגיל זה לצאת אל ערוץ הפלט הסטנדרטי באמצעות cout:
1. באם לא קיים קובץ שניתן לפקודה באמצעות המחרוזת <filename>, או אם לא קיימות הרשאות לפעולה הנדרשת

(למשל, ניסיון לכתוב לקובץ ריק במיקום גדול מאפס), על הודעת השגיאה לציין את שם הקובץ באופן הבא:

```
error: <filename> does not exist or cannot be processed.\n
```

2. באם לא קיימת פקודה שהוקדה ע"י המשתמש, יש להוציא את ההודעה הבאה:

```
error: command is not supported.\n
```

3. באם השימוש בפקודה הינו שגוי, על הודעת השגיאה לציין את פורמט הפקודה הנכון באמצעות המילה USAGE;

למשל, בהקלדת הפקודה tail ללא שם קובץ, להלן הודעת השגיאה המצופה:

```
error: USAGE tail FILENAME [n].\n
```

* שימו לב שניסיון לקרוא מקובץ ריק אינו נחשב לשגיאה – אלא פשוט לא יגרור פלט.

דוגמאות למקרי קצה/שגיאה:

[input]	touch test0
[input]	write test0 100 d
[output]	error: test0 does not exist or cannot be processed.
[input]	read test0 0
[input]	...

לאחר קומפילציה של התכנית לכדי קובץ הרצה בשם `cppEmulate`, הרצתו בטרמינל תפתח ערוץ האזנה לקלט חופשי מן המשתמש עבור מערכת הקבצים ביחס לתיקיית ההרצה; להלן דוגמא לרצף אפשרי של קלטים:

```
[terminal] $ ./cppEmulate
[input] touch test.txt
[input] write test.txt 0 d
[input] read test.txt 0
[output] d
[input] write test.txt 1 a
[input] write test.txt 2 n
[input] write test.txt 3 s
[input] write test.txt 4 k
[input] write test.txt 5 !
[input] cat test.txt
[output] error: test.txt does not exist or cannot be processed.
[input] cat test.txt
[output] dansk!
[input] exit
[terminal] $
```

דגשים

- במקרים של אי-תקינות בעבודה מול המשתמש (למשל, קבצים שאינם קיימים), או במקרה של שגיאות זמן-אמת, יש להפעיל מנגנון חריגות בשכבת אובייקט הקובץ, לתפוס אותן בשכבת האמולטור, ולהציג למשתמש הודעות שגיאה מתאימות בערוץ הפלט הסטנדרטי (`cout`). בכל מקרה, מצב בו חריגה איננה נתפסת ומביאה לקריסת התכנית תיחשב לשגיאה תכנותית!
- אתם רשאים להשתמש בספריית STL בתרגיל זה.
- אין להשתמש בפקודות מערכת לשם מימוש פקודות הקבצים (למשל, באמצעות פקודת `system`), אך אתם בהחלט מוזמנים לבסס את שלב הבדיקות שלכם, בטרם ההגשה, על פקודות המערכת.
- עליכם לוודא כי התכנית עוברת קומפילציית `g++` התואמת את הקומפיילר שעל שרת המכללה ללא כל שגיאות או אזהרות כלשהן, ורצה בהצלחה.
- תיתכנה התנהגויות מערכת שונות במימוש הפעולות על קבצים (למשל, מימוש פעולת `touch`); לכן, בתרגיל זה בפרט מומלץ לעבוד בסביבה מבוססת-יוניקס.
- עליכם לתעד את הקוד באמצעות הערות המתארות בקצרה את הפונקציות השונות.

הגשה

- עליכם להגיש קובץ ארכיב מטיפוס `zip` בלבד, ששמו כולל את קוד הקורס ('31'), שם התרגיל ('exX') ותעודת הזהות של הסטודנט/ית המגישה, מופרדים בקו תחתי בפורמט הבא: `31_exX_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 `*.h`, `*.cpp` ו-`*.txt`.
- לדוגמה: על סטודנט שמספר הזיהוי שלו הינו 012345678 להגיש ארכיב בשם `31_exX_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!