צ'ודנובסקי דמיטרי 324793900 קומפילציה 20364 א2021 ממ"ן 16

המתרגם תוכניות משפת CPL לשפת QUAD

1. **תיאור ומבנה הקוד:**

a. מימוש מבנה נתונים לפקודות QUAD:

נשתמש במבנה נתונים של רשימה מקושרת כדי לשמור את הפקודות של שפה Quad.ברגע ש- lexer ו- parser בודקים נכונות הקלט בקובץ ou. של כל שורה ונמצאה התאמה בין השורה הזאת לאחת הכללי הגזירה של ה- parser (של הדקדוק שניתן לראות בעמוד 37 בחוברת הקורס) הפונקציות של התוכנית הזאת יודעות לתרגם את שורה של השפה CPL לשורה של שפת Quad.אם נמצאה שגיאה באי נכונות של השורה בבדיקה של lexer ו- parser התוכנית תיתן הודעת שגיאה ומספר השורה איפה ששגיאה קרתה.כדי להשתמש במבנה נתונים באופן יעיל יותר אז לכל רשימה יוצרים תכונה שהיא מצביע לאיבר אחרון שלה כך שברגע שמחברים נגיד שתי רשימות לרשימה אחת,לא נצטרך לסרוק את כל הרשימה מהראש עד הסוף אלא מחברים איבר אחרון של רשימה ראשונה לראש של רשימה שנייה.ביצירת פקודה או רשימה של פקודות משתמשים ב- malloc כלומר בהקצאת זיכרון דינמית.

לגבי חישובי יעדי הקפיצה בשפה Quad,אנו יודעים שיש שני פקודות שהן JMPZ ו- JUMP כאשר יעד הקפיצה הוא מספר השורה.כדי להגיע למספר שורה נכונה נצטרך ליצור קוד זמני שמכיל תוויות סימבוליות ועוד מעבר נוסף על הקוד כדי להחליף את התוויות הסימבוליות במספרי שורות.כל תווית שומרת את מספר שורה שלה לכן פשוט במקומה רושמים את מספר השורה.ולכן בהדפסה נראה מספר השורה ולא שם של התווית.

b. מימוש מבנה נתונים לטבלת סמלים:

נשתמש במבנה נתונים של עץ חיפוש בינארי כדי לשמור את כל המשתנים שיש בתוכנית.בטבלה שומרים את שם המשתנה וסוג שלו וגם האם הוא חי או מת.אם הוא במצב FREE אז הוא מת ולכן אפשר לשחרר אותו ואם הוא SAVE אז הוא חי ונשתמש איתו בהמשך ואם הוא LOCK אז הוא חי ומשתמשים בו כרגע.את סוג המשתנה יכול להיות R או I.אם יש שני משתנים שמבצעים עליהם פעולה אז תוצאה תהיה הבא:

- ראשון R שני R תוצאה R
- ראשון I שני I תוצא I
- ראשון R שני I תוצאה R

במצב ההשמה כאשר ראשון מצד שמאל ושני מצד ימין:

- ראשון R שני R תוצאה R
- ראשון I שני I תוצא I
- ראשון R שני I תוצאה R
- ראשון I שני R תוצאה שגיאה (רק במקרה של CAST ניתן להעביר FLOAT ל- INT)

סידור משתנים בעץ מתבצע לפי סדר לקסיקוגרפי.נניח משתנה A יהי מצד שמאל של העץ לאומת זאת משתנה Z יהיה מצד ימין של העץ.זה נותן לנו חיפוש מהיר של משתנה בעץ ברגע שאנו צריכים למצוא אותו ולבצע עליו פעולה כלשהי.

צ'ודנובסקי דמיטרי 324793900 קומפילציה 20364 א2021 ממ"ן 16

2. **<u>רשימת קבצים:</u>**

a. <u>lexer.lex:</u> מנתח לקסיקלי של הקלט.

b. <u>lex.yy.c:</u> המימוש של מנתח לקסיקלי שנוצר על ידי $flex$.

c. <u>parser.y:</u> מנתח תחבירי וסמנטי של הקלט.

d. <u>parser.tab.c:</u> המימוש של מנתח תחבירי שנוצר על ידי $bison$.

e. <u>parser.tab.h:</u> המימוש של מנתח תחבירי שנוצר על ידי $bison$.

f. <u>cpq.c:</u> בקובץ הזה נמצא את ה- main שמטרתו לפתוח את קובץ הקלט ou. ליצור קובץ הפלט qud. , להפעיל על הקובץ הקלט $yyparse$ כלומר להתחיל את התהליך של התרגום ולהודיע האם תרגום התבצע בהצלחה.

g. <u>cpq.h:</u> קובץ עזר לקובץ cpq.c.

h. <u>symtab.c:</u>בקובץ הזה אנו מנהלים טבלת סמלים שלנו בעזרת מבנה נתונים עץ חיפוש בינארי.בקובץ נמצאות כל פונקציות לניהול הטבלה כמו הוספה, חיפוש,שחרור של משתנים וקביעת סטאטוס שלהם בתוכנה.

i. <u>symtab.h:</u> קובץ עזר לקובץ symtab.c.

j. <u>gen.c:</u> מטרתו של קובץ הזה הוא בעצם לבצע תרגום של שפת CPL לשפת QUAD.המבנה נתונים בנוי כרשימה מקושרת ואוספת שורות של פקודות מקובץ ou. כך אנו מקבלים שרשרת של שורות וכל שורה מתרגמים לשפה QUAD ורושמים את תוצאה לקובץ qud..בקובץ נמצא כל הניהול על רשימה מקושרת כמו לחבר שתי פקודות או לחבר רשימה של פקודות לפקודה וכו'.

k. <u>gen.h:</u> קובץ עזר לקובץ gen.c.