האלגוריתם:

1. מאתחלים את הIC והDC ל0
2. קוראים שורה מקובץ המקור – אם הקובץ נגמר לדלג לשלב 17.
3. אם השורה ריקה או קומנט, לחזור לשלב 2.
4. השורה הגדרת קבוע? אם לא לך לשלב 6.
5. הוסף את הקבוע לטבלה עם המאפיין mdefine, חזור לשלב 2.
6. השורה הגדרת תווית? אם לא עבור לשלב 8.
7. הדלג דגל "יש תווית"
8. זו בקשה לשמור מידע (.data, .string)? אם לא עבור לשלב 11.
9. אם יש הגדרת תווית, הוסף אותה לטבלה עם מאפיין data, ערכה DC.
10. מצא את סוג הגדרת המידע וקודד אותו לזיכרון, עדכן את DC לפי אורך המידע. אם ההגדרה היא .data ויש לה שימוש בקבוע, וודא שהקבוע בטבלה עם המאפיין mdefine. חזור לשלב 2.
11. האם זוהי בקשה לשמירת קוד (.entry, .extern)? אם לא עבור ל13.
12. האם זה .extern? אם כן, הוסף כל אחד מהסמלים שמופיעים כאופרנדים לטבלה בלי ערך ועם מאפיין external חזור לשלב 2.
13. זו הגדרת תווית? אם כן, הוסף אותה לטבלה עם מאפיין code וערכה IC+100
14. מצא את שם האופרציה בטבלת האופרציות וקודד אותה לזיכרון.
15. נתח את מבנה האופרנדים וחשב את L (כמות המילים שהפקודה לקחה). בנה את הקוד הבינארי של המילה הראשונה של האופרציה וקודד אותה לזיכרון. אם יש עוד מילה, קודד גם אותה.
16. עדכן את IC בL. חזור לשלב 2.
17. הקובץ נקרא, אם היו שגיאות עצור כאן.
18. עדכן את כל הטבלה כך שלכל מי שיש מאפיין data הוסף לערכו IC+100.
19. התחל את המעבר השני.

יישום באמצעות קוד:

במקרה של .data נתרגם כל אחד מהמספרים לבינארי ונכתוב אותם לקובץ הפלט.

במקרה של .string נתרגם כל אחד מהתווים לקוד אסקי ואז לבינארי ונכתוב אותם לקובץ הפלט.

במקרה של .define נשמור את הקבוע בטבלה.

במקרה של .extern נשמור כל אחת מהתוויות לטבלה.

במקרה של פקודה:

יהיו לנו שני מבנים.

הראשון יהיה לopcode, למבנה יהיה מחרוזת שמכילה את הבינארי של הopcode ומספר שמכיל את כמות המספרים שהפקודה מקבלת. נכיל מערך שלם של המבנה הזה לכל אחת מהפקודות וכך יהיה לנו את כל הפקודות שמורות במערכת, ברגע שניתקל בפקודה נחפש את שמה במערך המבנים וכך נוכל להשיג את קודה.

המבנה השני יהיה לחלקים של הפקודה. למבנה יהיה את הopcode של הפקודה כמחרוזת, את אופרנד המקור כמחרוזת ואת אופרנד היעד כמחרוזת.

כאשר נתקל בפקודה נחפש את ערכה במערך האופקודים, לאחר שנשיג אותו נקרא לפונקציה שתיצור משתנה מסוג המבנה של החלקים שתכיל את כל מה שאנו צריכים כדי לתרגם את הפקודה, הפונקציה תחזיר את המבנה מוכן ובפונקציה שקראה לה נחבר את כל הערכים ששמורים במשתנה למחרוזת אחת.

לפי המשתנה של כמות הארגומנטים שנמצא במערך האופקודים נדע כמה משתנים אנו צריכים לתרגם בנוסף לפקודה המרכזית, תהיה לנו פונקציה שמקבלת ארגומנט ומתרגמת אותו לשורה בינארית, במקרה ואלו שניהם רגיסטרים תהיה פונקציה נפרדת שמטפלת במקרה הספציפי הזה.

הטבלה תנוהל באמצעות אותו המבנה שהשתמשנו בו כדי לנהל את המקרואים, גם לו היה name, type, value ולכן אין צורך במבנה נוסף.