НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
Кафедра системного програмування і спеціалізованих комп'ютерних систем

ПОЯСНЮВАЛЬНА ЗАПИСКА

з дисципліни «Системне програмування»
на тему «Розробка компілятора програм мовою Асемблера»

Студента _____ курсу _____ групи
напряму підготовки
6.050102 «Комп'ютерна інженерія»

_____
(прізвище та ініціали)

Керівник _____
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Національна оцінка _____

Кількість балів: _____ Оцінка: ECTS _____

Члени комісії _____ _____
(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

_____ _____
(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

_____ _____
(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

КИЇВ – 2014

ТЕХНІЧНЕ ЗАВДАННЯ

1. Вхідні дані транслятора - текстовий файл з довільною програмою на мові Асемблера, складеною в відповідності з обмеженнями, які задані в варіанті курсової роботи. Для підготовки програми на мові Асемблера використовується, наприклад, стандартний додаток OS Windows Блокнот.
2. На всі синтаксичні конструкції (ідентифікатори, константи, директиви, машинні команди, режими адресації і т.д.), які допускаються в TASM(MASM) і які виходять за рамки обмежень в варіанті курсової роботи повинно видаватись діагностичне повідомлення про синтаксичну помилку.
3. В результаті роботи транслятора повинен бути створений текстовий файл лістінга (розширення .lst). Формат файлу лістінга повинен співпадати з форматом файлу лістінга MASM або TASM. Діагностичні повідомлення формуються на українській мові. Таблиця символів в файлі лістінга може бути в довільному форматі.
4. Транслятор повинен аналізувати командний рядок, в якому задаються імена початкового файлу та файлу лістінга. Всі діагностичні повідомлення, які формуються в файлі лістінга додатково повинні виводитись на екран монітора. Крім того, на екран виводиться загальна кількість помилок, виявлених в початковій програмі.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

Ідентифікатори

Містять великі і малі букви латинського алфавіту та цифри. Починаються з букви. Великі та малі букви не відрізняються. Довжина ідентифікаторів не більше 8 символів

Константи

Шістнадцяткові, десяткові та двійкові константи

Директиви

END,

SEGMENT - без операндів, ENDS, програма може мати тільки один сегмент кодів і тільки один сегмент даних

DB,DW,DD з одним операндом - довільний арифметичний вираз над константами

Розрядність даних та адрес

32 - розрядні дані та зміщення в сегменті, 16 - розрядні дані та зміщення не використовуються

Адресація операндів пам'яті

Індексна адресація (Val1[eax],Val1[edi] і т.п.)

Заміна сегментів

Префікси заміни сегментів можуть задаватись явно, а при необхідності автоматично генеруються транслятором

Машинні команди

Movsb

Rep Movs mem

Add **reg,reg**

Or **reg,mem**

Test **mem,reg**

Mov **reg,imm**

Shl **mem,imm**

Jnz,

де **reg –** 8 або 32-розрядні РЗП;

mem – адреса операнда в пам’яті;

imm – 8 або 32-розрядні безпосередні дані (довільний арифметичний вираз над константами).

ТЕСТОВІ ФАЙЛИ

Без помилок для створюваного компілятора

```
DATA  SEGMENT
            ID1         DB    255
            ID2         DW    1111B
            iD3         DD    5BC35FF6H

            AR1         DD    10
            AR2         DD    10
DATA  ENDS

CODE  SEGMENT
      BEGIN:
            ; AND test
            MOV         AL, 0FH
            MOV         BL, 11H + 00001111B * 13
            ADD         AL, BL
            ADD         EBX, ECX

            ; OR test
            MOV         CL, 00H
            OR          CL, ID1[EDI]
            JNZ         LABEL_F

            ; MOV test
            MOV         EAX, 0FFFFFFFH
            MOV         BL, 0AH

            ; TEST test
            TEST  iD3[ESI], 147
            TEST  CS:iD3[EDI], 5H

      LABEL_F:
            ; SHL test
            SHL         ID3[ESI], 00000001B
            SHL         CS:ID1[ESI], 11H

            ; JNZ TEST
            JNZ         BEGIN

            ; SEGMENT change
            TMP1    DB 13
            TEST  TMP1[EDI], AL
            TMP2  DW    0334H
```

```
            SHL         TMP2[EDI], 3
            TMP3  DD    43444546H
            TEST  TMP3[ESI], EAX


            MOV         ECX, 10
            REP         MOVS  ID2[EDI], ID3[ESI]
CODE  ENDS

            END   BEGIN
```

Без помилок для MASM

```
.386

DATA  SEGMENT
            ID1         DB    255
            ID2         DW    1111B
            iD3         DD    5BC35FF6H

            AR1         DD    10
            AR2         DD    10
DATA  ENDS

CODE  SEGMENT
            ASSUME DS:DATA, CS:CODE

      BEGIN:
            ; AND test
            MOV         AL, 0FH
            MOV         BL, 11H + 00001111B * 13
            ADD         AL, BL
            ADD         EBX, ECX

            ; OR test
            MOV         CL, 00H
            OR          CL, ID1[EDI]
            JNZ         LABEL_F

            ; MOV test
            MOV         EAX, 0FFFFFFFH
            MOV         BL, 0AH

            ; TEST test
            TEST  iD3[ESI], 147
```

```
            TEST  CS:iD3[EDI], 5H

      LABEL_F:
            ; SHL test
            SHL         ID3[ESI], 00000001B
            SHL         CS:ID1[ESI], 11H

            ; JNZ TEST
            JNZ         BEGIN

            ; SEGMENT change
            TMP1    DB 13
            TEST  TMP1[EDI], AL
            TMP2  DW    0334H
            SHL         TMP2[EDI], 3
            TMP3  DD    43444546H
            TEST  TMP3[ESI], EAX


            MOV         ECX, 10
            REP         MOVS  ES:ID2[EDI], ID3[ESI]
CODE  ENDS

            END   BEGIN
```