НАЦІОНАЛЬНИЙ ТЕХНІЧНЙ УНІВЕРСИТЕТ УКРАЇНИ

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

*Звіти до комп’ютерних практикумів дисципліни*

«Системне програмне забезпечення»

**Прийняв Виконав**

**доцент кафедри ІПІ Студент групи ІП-24**

**Лісовиченко О.І. Піддубний Б.С.**

**“12” лютого 2024р.**

Київ 2024

**Комп’ютерний практикум №1**

**Тема:** Створення програм на асемблері

**Завдання:**

1. Для програми, наведеної вище, створити файл типу .asm. Ця програма не має засобів виводу даних, тому правильність її виконання треба перевірити за допомогою td.exe.
2. Скомпілювати програму, включивши потрібні опції для налагоджувача та створення файлу лістингу типу .lst.
3. Ознайомитись зі структурою файлу .lst. За вказівкою викладача, для певної команди асемблера розглянути структуру машинної команди і навести її у звіті.
4. Скомпонувати .obj-файл програми. Включити опції для налагодження та створення .map-файлу.
5. Занести до звіту адреси початку та кінця всіх сегментів з .map-файлу.
6. Завантажити до налагоджувача td.exe одержаний .exe-файл програми.
7. У вікні CPU у полі DUMP знайти початкову адресу сегмента даних та записати його до звіту. Знайти масиви SOURCE та DEST. Дані у масиві SOURCE подаються у шістнадцятковій системі.
8. У покроковому режимі за допомогою клавіші F7 виконати програму. Одержані результати у масиві DEST показати викладачеві.

Текст програми:

; Опис сегменту стеку

STSEG SEGMENT PARA STACK "STACK"

DB 64 DUP ( "STACK" )

STSEG ENDS

; Опис сегменту даних

DSEG SEGMENT PARA PUBLIC "DATA"

SOURCE DB 10, 20, 30, 40

DEST DB 4 DUP ( "?" )

DSEG ENDS

; Опис сегменту коду

CSEG SEGMENT PARA PUBLIC "CODE"

; Код основної функції

MAIN PROC FAR

ASSUME CS: CSEG, DS: DSEG, SS: STSEG

; Адреса повернення

PUSH DS

MOV AX, 0 ; або XOR AX, AX

PUSH AX

; Ініціалізація DS

MOV AX, DSEG

MOV DS, AX

; Обнулення масиву

MOV DEST, 0

MOV DEST+1, 0

MOV DEST+2, 0

MOV DEST+3, 0

; Пересилання

MOV AL, SOURCE

MOV DEST+3, AL

MOV AL, SOURCE+1

MOV DEST+2, AL

MOV AL, SOURCE+2

MOV DEST+1, AL

MOV AL, SOURCE+3

MOV DEST, AL

RET

MAIN ENDP

CSEG ENDS

END MAIN

**Введені та отримані результати**

**Вміст .lst файлу:**

**Вміст .map файлу:**

**Вікно DUMP:**

**До виконання програми:**

**Масив SOURCE:**

**Масив DEST:**

**Після виконання програми:**

**Масив DEST:**

**Висновок:**

1. В текстовому редакторі було створено файл типу .asm.
2. Скомпілював програму, включивши потрібні опції для налагоджувача та створення файлу лістингу типу .lst.
3. Ознайомився зі структурою файлу .lst. Розглянув структури машинних команд.
4. Після усунення помилок, скомпонував .obj-файл програми, включивши опції для налагодження та створення .map-файлу.
5. Відкрив файл карти памяті (.map-файл) та подивився на адреси початку та кінця всіх сегментів програми.
6. Завантажив програму налагоджувача td.exe та мій одержаний .exe-файл програми.
7. У вікні CPU у полі DUMP подивився на початкову адресу сегмента даних. В сегменті даниз знайшов масиви SOURCE та DEST. Дані у масиві SOURCE подаються у шістнадцятковій системі.
8. У покроковому режимі за допомогою клавіші F7 виконав програму. Програма коректно виконує поставлену задачу.