**PHẦN 1: PHẦN LÀM CÁ NHÂN**

**Vương Đức Trọng : B22DCCN866**

**1.1.1 Bài số 1: Lập trình hợp ngữ Assembly:**

**Câu 3:** Viết chương trình hợp ngữ Assembly cho phép nhập 1 chuỗi ký tự và in ra màn hình chuỗi ký tự đó.

A screenshot of a cell phone

Description automatically generated

**Hình 1: Flow** **chart** “nhập 1 chuỗi ký tự và in ra màn hình chuỗi ký tự đó”

**A screenshot of a computer

Description automatically generated**

**Hình 2: Mã nguồn “**nhập 1 chuỗi ký tự và in ra màn hình chuỗi ký tự đó**”**

**A screen shot of a computer

Description automatically generated**

**Hình 3: Giao diện hiển thị “**nhập 1 chuỗi ký tự và in ra màn hình chuỗi ký tự đó**”**

**Câu 12**: Viết chương trình hợp ngữ Assembly cho phép nhập vào một số và in ra màn hình giai thừa của số đó.

A screenshot of a computer

Description automatically generated

**Hình 4: Flow chart “**nhập vào một số và in ra màn hình giai thừa của số đó**”**

**A screenshot of a computer

Description automatically generated**

**Hình 5: Mã nguồn “**nhập vào một số và in ra màn hình giai thừa của số đó**”**

**A screenshot of a computer

Description automatically generated**

**Hình 6: Giao diện hiển thị “**nhập vào một số và in ra màn hình giai thừa của số đó**”  
  
   
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
Câu 18:** Viết chương trình hợp ngữ đếm số lần xuất hiện của chuỗi con "ktmt"trong một chuỗi. In kết quả dưới dạng số thập phân.

A screenshot of a computer

Description automatically generated

**Hình 7: Flow chart “**đếm số lần xuất hiện của chuỗi con "ktmt"trong một chuỗi**”**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Hình 8: Mã nguồn “**đếm số lần xuất hiện của chuỗi con "ktmt"trong một chuỗi**”**

**A computer screen with white text

Description automatically generated**

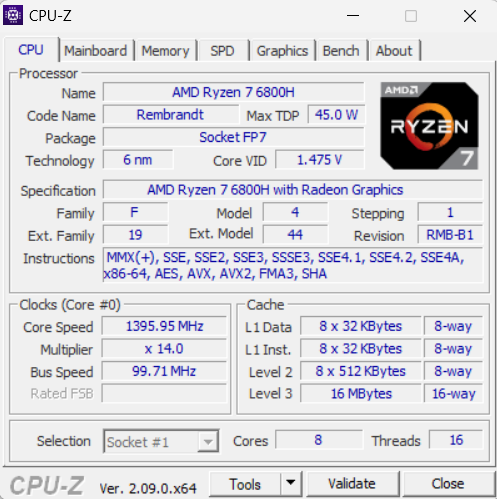
**Hình 9: Giao diện hiển thị “**đếm số lần xuất hiện của chuỗi con "ktmt"trong một chuỗi**”**

**1.1.2. Bài số 2: Thực hành phân tích khảo sát bộ nhớ:**

1. Khảo sát cấu hình của máy và hệ thống bộ nhớ của máy đang sử dụng (Bộ nhớ trong: ROM, RAM, Cache System, Bộ nhớ ngoài: ổ đĩa cứng, CD, Thiết bị vào ra.)

Khảo sát cấu hình của máy và hệ thống bộ nhớ của máy đang sử dụng Sử dụng phần mềm CPU-Z 64-bit Ver.2.09.0x64:

CPU:

****

Cache:

A screenshot of a computer

Description automatically generated

RAM:

A screenshot of a computer program

Description automatically generated

ROM:

A screenshot of a computer

Description automatically generated

Bộ nhớ ngoài: DVD

A close up of a dvd

Description automatically generated

Thiết bị vào ra:

A screenshot of a computer

Description automatically generated

2. Dùng công cụ Debug khảo sát nội dung các thanh ghi IP, DS, ES,

SS, CS, BP, SP

Công cụ sử dụng: emu8086 microprocessor emulator

Các bước thực hiện:

• Mở file .asm bằng phần mềm trên

• Chọn emulate trên thanh công cụ rồi chọn nút debug nằm cuối của cửa sổ vừa mở ra

• Chạy Single step để xem kết quả debug từng mã lệnh từ đầu đến cuối

A screenshot of a computer

Description automatically generated

3. Giải thích nội dung các thanh ghi, trên cơ sở đó giải thích cơ chế quản lý bộ nhớ của hệ thống trong trường hợp cụ thể này.

A.Nội dung các thanh ghi:

1. Các Thanh ghi Đoạn (Segment Registers):

+ CS (Code Segment): Thanh ghi đoạn mã. CS chứa địa chỉ bắt đầu đoạn mã.

+ DS (Data Segment): Thanh ghi đoạn dữ liệu. DS chứa địa chỉ bắt đầu đoạn dữ liệu.

+ SS (Stack Segment): Thanh ghi đoạn ngăn xếp. SS chứa địa chỉ bắt đầu đoạn ngăn xếp.

+ ES (Extra Segment): Thanh ghi đoạn dữ liệu mở rộng. ES chứa địa chỉ bắt đầu đoạn dữ liệu mở rộng.

2. Các Thanh ghi Con trỏ (Pointer Registers):

+ IP (Instruction Pointer): Con trỏ lệnh (còn gọi là bộ đếm chương trình PC). IP luôn chứa địa chỉ của lệnh tiếp theo sẽ được thực hiện.

• Thanh ghi IP (con trỏ lệnh) chứa địa chỉ offset của lệnh tiếp theo.

• CS:IP chứa địa chỉ logic của lệnh tiếp theo đó.

+ SP (Stack Pointer): con trỏ ngăn xếp. SP luôn chứa địa chỉ đỉnh ngăn xếp SS:SP.

+ BP (Base Pointer): Con trỏ cơ sở, chứa địa chỉ của dữ liệu trong đoạn ngăn xếp SS hoặc các đoạn khác SS:BP.

B. Quản lý bộ nhớ cụ thể trong chương trình này

1.Khởi tạo DS:

Lệnh: MOV AX, @Data :

Tải địa chỉ cơ sở của đoạn dữ liệu (0720h) vào thanh ghi AX.

Lệnh: MOV DS, AX :

Gán giá trị của AX (0720h) vào DS, thiết lập DS trỏ đến đoạn dữ liệu.

2.Thiết lập và sử dụng ngăn xếp:

Ngăn xếp:

Đoạn ngăn xếp bắt đầu tại địa chỉ 07100h với SP=0100, nghĩa là đỉnh ngăn xếp tại địa chỉ 07100h+0100h.

3.Gọi hàm in chuỗi ký tự:

Lệnh: MOV AH, 09h:

Thiết lập AH với mã hàm 09h của ngắt 21h, dùng để in chuỗi ký tự.

Lệnh: LEA DX, HAM1:

Tải địa chỉ hiệu dụng của chuỗi "Hello world !$" vào DX. Trong trường hợp này, địa chỉ hiệu dụng là 0003h.

Lệnh: INT 21H:

Gọi ngắt 21h để thực hiện in chuỗi ký tự.

=> Cơ chế quản lý bộ nhớ trong hệ thống vi xử lý 8086 dựa trên việc sử dụng các thanh ghi đoạn để phân chia và quản lý các đoạn bộ nhớ khác nhau cho mã, dữ liệu, và ngăn xếp. Trong chương trình cụ thể này, đoạn mã (CS) chứa các lệnh thực thi, đoạn dữ liệu (DS) chứa các biến và chuỗi ký tự, và đoạn ngăn xếp (SS) chứa dữ liệu tạm thời và địa chỉ trả về. Các thanh ghi con trỏ như IP, SP, và BP giúp định vị và quản lý các lệnh và dữ liệu trong các đoạn tương ứng.