|  |  |  |  |
| --- | --- | --- | --- |
|  | **Ho Chi Minh City University of Technology**  **Department of Electrical and Electronics Engineering** | | |
| **FINAL EXAMINATION**  Grading: 40% | | | **Computer System Engineering**  Course ID: 407406 |
| **Date: 14 Dec., 2018** | | | **Duration:** 90 minutes |
| **Student name:**  **Student ID:** | | | **Examiner’s name & signature:** |
| **Score:** | | Students are allowed to use *one A4 page with two sides* for reference.  Books and other documents are not allowed to use. | |
| **This examination consists of 4 pages** | |

**Problem 1:** (20pts) Answer the following questions

1. Which technique allows programs to address more than 64 KB memory?

|  |
| --- |
| **Segmentation** |

1. How many bits of external address are there in 8086 processor?

|  |
| --- |
| **20 bits of external address** |

1. Which registers belong to Core-i5 processor among the following registers: AX, BX, CX, EAX, EBX, ECX, RAX, RBX, RCX?

|  |
| --- |
| **RAX, RBX, RCX** |

1. Find the five-hex-digit address that corresponds to each of these segment : offset pairs

2B87:836A => 33BDA

56CD:B24E => 61F1E

1. Assume that we have the memory content as below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Address | 0x0 | 0x1 | 02 | 0x3 |
| Content | 62 | 5F | C3 | 2F |

What are the 32-bit data when we read a double-word at the address 0x0 with Big Endian mode? **625FC32F**

**Problem 2:** (20pts) Answer the value of registers after the instruction is executed.

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Before** | **Instruction** | **After** |
| 1 | AX: 25 A4  BX: FF 34 | mov bx, ax | AX: 25 A4  BX: 25 A4 |
| 2 | ECX: 0A 76 FF FF | inc ecx | ECX: 0A 77 00 00 |
| 3 | EAX: 3A 54 12 4C  EBX: 00 00 00 04 | mul ebx | EAX: E9 50 49 30  EBX: 00 00 00 04 |
| 4 | EAX: 00 00 00 78 | sub eax, 120 | EAX: 00 00 00 00  SF:0 ZF:1 CF:0 OF:0 |
| 5 | AX: 03 10  word at Value: 01 F2 | imul ax, Value | AX: F5 20  SF:0 ZF:0 CF:0 OF:1 |

**Problem 3:** (10pts) Write 80x86 assembly language code for the following C procedure:

|  |  |
| --- | --- |
| **C procedure** | **ASM procedure**  *Assume that S is stored in EAX, N is stored in EBX* |
| int factorial(int N)  {  int i;  int S=1;  if (N ==0)  S = 1;  else{  for (i=1; i<=N; i++)  {  S = S\*i;  }  }  return S;  } | **factorial:**  **mov EAX, 1**  **cmp EBX, 0**  **je EXIT**  **mov ECX, EBX**  **FOR\_LOOP:**  **cmp ECX, 0**  **je EXIT**  **mul EAX, ECX**  **dec ECX**  **jmp FOR\_LOOP**  **EXIT:**  **ret** |

**Problem 4:** (10pts) Write 80x86 assembly language code for the following C function.

|  |  |
| --- | --- |
| int arith(int x, int y, int z)  {  int t1 = x+z;  int t2 = y + t1;  int t3 = x+8;  int t4 = y \* 15;  int t5 = t1 + t4;  int rval = t3 \* t5;  return rval;  } | **arith:**  **(x:EAX , y:EBX , z:ECX)**  **add ECX,EAX**  **add ECX,EBX**  **add EAX,8**  **imul EBX,15**  **add EAX,EBX**  **imul ECX, EAX** |

**Problem 5:** (10pts) Write an 80x86 Assembly language program to compute S = 54\*(x+y) - 49\*z + 8. Assume that:

* S is stored in register EAX
* x is stored in register EBX
* y is stored in register ECX
* z is stored in register EDX

|  |
| --- |
| **add EBX,ECX**  **imul EBX,54**  **imul EDX,49**  **sub EBX,EDX**  **add EBX,8**  **mov EAX,EBX** |

**Problem 6:**  (10pts) Given the Interrupt Vector Table below.



Determine the address of ISR of a device with the interrupt vector FBh.

|  |
| --- |
| Address in table = 4 X FBH = 3ECH  (Multiply by 4 since each entry is 4 bytes)  Offset Low = [3EC] = 3A, Offset High = [3ED] = 54  Segment Low = [3EE] = 54, Segment High = [3EF] = 7F  Address = 7F54:543A = 7F540 + 543A = 8497A |

**Problem 7:**  (10pts) What are the purposes of privilege levels ?

|  |
| --- |
| **The purposes of privilege levels are to prevent:**   * **Users from interfering with one another** * **Users from examining secure data** * **Program bugs from damaging other programs** * **Program bugs from damaging data** |

**Problem 8:** (10pts)

1. Write C++ instructions to free the memory of the following array:

Student \*\*p = new Student\*[500];

for(int i=0;i<500;i++)

p[i] = new Student[50];

|  |
| --- |
| for(int i=0; i<500; i++)  delete p[i];    delete[] p; |

1. Write C++ instructions to provide memory allocation for 200 float numbers which are addressed by the pointer q.

float \*q = new float[200];

*--------------------------------------------------- The end ------------------------------------------------------*

*Electronics Department Lecturer*

*Dr. Tran Hoang Linh Dr. Truong Quang Vinh*