CPSC 240: Computer Organization and Assembly Language

Assignment 03, Fall Semester 2023

CWID:\_\_885857847\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Name:\_Kush Patel\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Download the “CPSC-240 Assignment03.docx” document.
2. Design the "multiplication.asm" program, and use assembly language to realize the function of the following C++ instructions.  
    unsigned int num1 = 300,000;  
    unsigned int num2 = 400,000;  
    unsigned long product = 0;  
    product = long(num1 \* num2);
3. Assemble the "multiplication.asm" file and link the "multiplication.o" file to get the "multiplication" executable file.
4. Run the "multiplication" file with the DDD debugger to display the simulation results of num1 and num2, as well as the simulation results of product.
5. Insert source code (multiplication.asm) and simulation results (GDB panel) of the memory (num1, num2, and product) in the document. Use calculator or hand calculation to verify simulation results.
6. Design the "division.asm" program, and use assembly language to realize the function of the following C++ instructions.  
    unsigned long num1 = 50,000,000,000;  
    unsigned int num2 = 3,333,333;  
    unsigned int quotient = 0, remainder = 0;  
    quotient = num1 / num2;  
    remainder = num1 % num2;
7. Assemble the "division.asm" file and link the "division.o" file to get the "division" executable file.
8. Run the "division" file with the DDD debugger to display the simulation results of num1 and num2, as well as the simulation results of quotient and remainder.
9. Insert source code (division.asm) and simulation results (GDB panel) of the memory (num1, num2, quotient, and remainder) in the document. Use calculator or hand calculation to verify simulation results.
10. Save the file in pdf format and submit the pdf file to Canvas before 23:59 pm on 09/20/2023.

[Insert multiplication assembly source code here]

; ex3\_multiplication1.asm

;unsigned int num1 = 300,000;

;unsigned int num2 = 400,000;

;unsigned long product = 0;

;product = long(num1 \* num2);

section .data

        num1     dd      300000                                     ;num1 = 25 = 19h

        num2     dd      400000                                     ;num2 = 35 = 23h

        mult     dq      0                                      ;mult = 0000h

section .text

        global \_start

\_start:

        mov eax, dword[num1] ;al = num1 = 19h

        mul dword[num2]

        mov dword[mult + 0], eax ;mult = ax = 036Bh

        mov dword[mult+4], edx

        mov     rax, 60                                         ;terminate excuting process

        mov     rdi, 0                                          ;exit status

        syscall                                                 ;calling system services

                                            ;calling system services

[Insert multiplication simulation results (GDB panel) here]

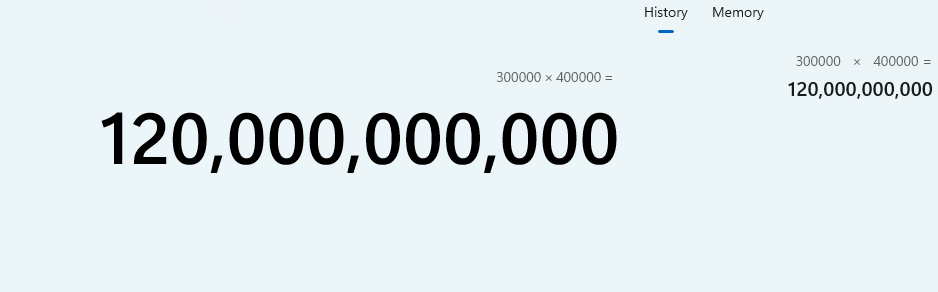
A screenshot of a computer program

Description automatically generatedA screen shot of a computer

Description automatically generatedA white rectangular object with a white background

Description automatically generatedA computer screen with a white screen

Description automatically generated

[Insert verification of calculator result or hand calculation here]

[Insert division assembly source code here]

section .data

        num1     dq      50000000000                                    ;num1 = 25 = 19h

        num2     dd      3333333;                                     ;num2 = 35 = 23h

        quotient  dd      0                                     ;quotient = 0000h

        remainder dd      0                                     ;remainder = 0000h

section .text

        global \_start

\_start:

        mov edx, dword[num1 + 4] ;al = num1 = 19h

         mov eax, dword[num1 + 0] ;al = num1 = 19h

        div dword[num2]

        mov dword[quotient], eax ;mult = ax = 036Bh

        mov dword[remainder], edx

        mov     rax, 60                                         ;terminate excuting process

        mov     rdi, 0                                          ;exit status

        syscall                                                 ;calling system services

                                            ;calling system services

[Insert division simulation results (GDB panel) here]

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA white rectangular object with a black border

Description automatically generatedA white rectangular object with a black border

Description automatically generatedA white rectangular object with a black border

Description automatically generated

[Insert verification of calculator result or hand calculation here]

Screenshot of a screenshot of a number of numbers

Description automatically generated