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]

```
 1 ; ex3_multiplication1.asm
 2 ;unsigned int num1 = 300,000;
 3 ;unsigned int num2 = 400,000;
 4 ;unsigned long product = 0;
 5 ;product = long(num1 * num2);
 6
 7 section .data
 8         num1     dd      300000                                  ;num1 = 25 = 19h
 9         num2     dd      400000                                  ;num2 = 35 = 23h
10         mult     dq      0                                     ;mult = 0000h
11
12 section .text
13         global _start
14
15 _start:
16         mov eax, dword[num1] ;al = num1 = 19h
17         mul dword[num2]
18         mov dword[mult + 0], eax ;mult = ax = 036Bh
19         mov dword[mult+4], edx
20
21         mov     rax, 60                                         ;terminate excuting process
22         mov     rdi, 0                                          ;exit status
23         syscall                                                 ;calling system services
24                                          ;calling system services
```

```
Breakpoint 1, _start () at multiplication.asm:21
(gdb) x/ud &num1
0x402000:       300000
(gdb)
```

```
Breakpoint 1, _start () at multiplication.asm:21
(gdb) x/ud &num1
0x402000:       300000
(gdb)
```

0x402000: 300000

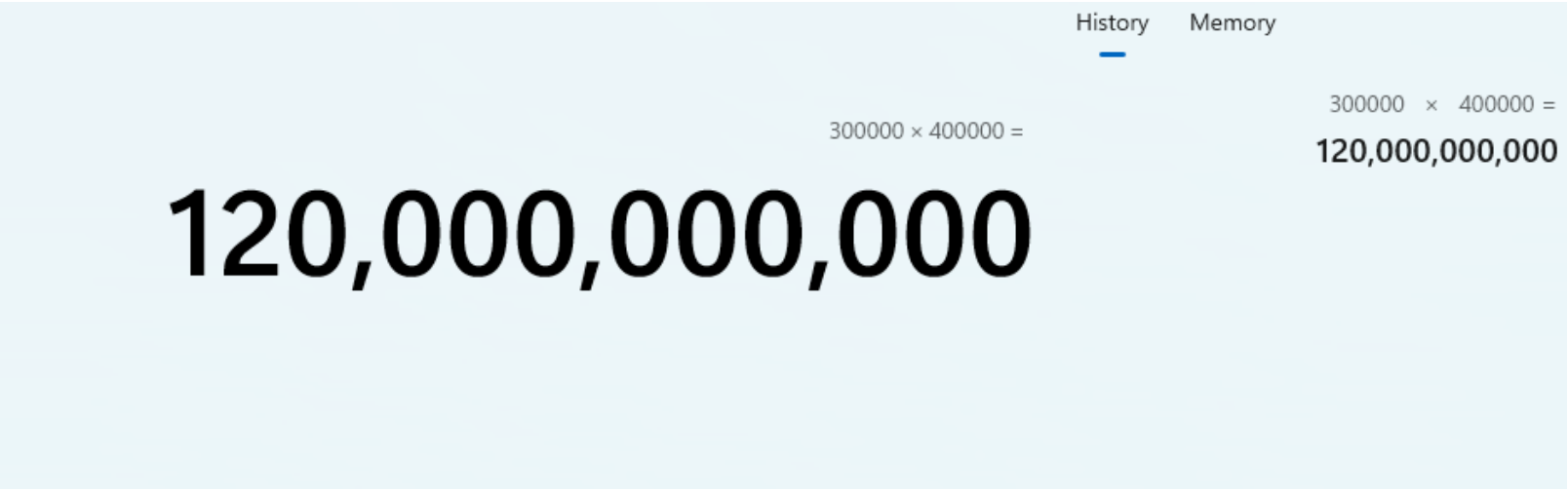
```
(gdb) x/ud &num2
0x402004:       400000
(gdb) x/ug &mult
0x402008:       120000000000
(gdb)
```

```
0x402008: 120000000000
```

```
Breakpoint 1, _start () at multiplication.asm:21
(gdb) x/ud &num1
0x402000:       300000
(gdb) x/ud &num2
0x402004:       400000
```

```
0x402008: 120000000000
```

[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]

```
 1
 2
 3 section .data
 4         num1      dq      50000000000                                   ;num1 = 25 = 19h
 5         num2      dd      3333333;                                    ;num2 = 35 = 23h
 6         quotient  dd       0                                       ;quotient = 0000h
 7         remainder dd       0                                       ;remainder = 0000h
 8
 9 section .text
10         global _start
11
12 _start:
13         mov edx, dword[num1 + 4] ;al = num1 = 19h
14          mov eax, dword[num1 + 0] ;al = num1 = 19h
15         div dword[num2]
16         mov dword[quotient], eax ;mult = ax = 036Bh
17         mov dword[remainder], edx
18
19
20
21
22         mov     rax, 60                                               ;terminate excuting proces
23         mov     rdi, 0                                                ;exit status
24         syscall                                                       ;calling system services
25                                             ;calling system services
26
```

```
Breakpoint 1, _start () at division.asm:22
(gdb) x/ug &num1
0x402000:       50000000000
(gdb)
```

0x402000: 50000000000

```
Breakpoint 1, _start () at division.asm:22
(gdb) x/ug &num1
0x402000:       50000000000
(gdb)
```

0x402000: 50000000000

```
(gdb) x/ug &num1
0x402000:       50000000000
(gdb) x/uw &num2
0x402008:       3333333
(gdb)
```

```
0x402008: 3333333
```

```
(gdb) x/uw &num2
0x402008:       3333333
(gdb) x/uw &quotient
0x40200c:       15000
(gdb)
```

```
0x40200c: 15000
```

```
(gdb) x/uw &quotient
0x40200c:       15000
(gdb) x/uw &remainder
0x402010:       5000
(gdb)
```

```
0x402010: 5000
```

[Insert verification of calculator result or hand calculation here]