

CPSC 240: Computer Organization and Assembly Language

Assignment 06, Spring Semester 2023

CWID: [886880327](#) Name: [Andrew Saldana](#)

- Download the "CPSC-240 Assignment06.docx" document.
- Design the "print.asm" program to calculate the sum of "1+2+3+...+100" and displays the result in a terminal window.
Calculates 1+2+3+...+100 and displays the result in a terminal window

```
char str1[] = "1+2+3+...+100=";  
register char cx = 1;  
short sum = 0;  
char ascii[5] = "0000\n";  
for(cx=1; cx<=100; cx++)  
    sum += cx;  
ascii = itoa(sum);  
cout << str1 << ascii;
```
- Assemble the "print.asm" file and link the "print.o" file to get the "print" executable file.
- Run the "print" file to display the conversion results of **ascii** in Terminal Emulator window.
- Insert source code (print.asm) and simulation results (Terminal Emulator window) at the bottom of the document.
- Save the file in pdf format and submit the pdf file to Canvas before 23:59 pm on 03/26/2023.

[Insert print.asm here]

[illegible]

```

53 ;getting the 4th digit
54 mov     ax, dx
55 cwd
56 mov     bx, 1
57 div     bx
58 add     byte[ascii+3], al
59
60 ; cout << str1
61 mov     rax, 1
62 mov     rdi, 1
63 mov     rsi, str1
64 mov     rdx, 24
65 syscall
66
67 ; cout << ascii
68 mov     rax, 1
69 mov     rdi, 1
70 mov     rsi, ascii
71 mov     rdx, 5
72 syscall
73
74 ;exit call
75 mov rax, 60
76 mov rdi, 0
77 syscall
78

```

```

;ax = dx = 0
;ax = ax:dx (changes 16 bit to 32 bit register)
;bx = 1
;ax = 0/1 = 0 ;bx = 0%1 = 1
;ascii+3 = al + 30h = 0 + 30h = 30h = '0'
;ascii = '5050', 10

;SYS_write
;where to write
;address of str1
;24 character to write
;calling system services

;SYS_write
;where to write
;address of ascii
;5 character to write(including newline)
;calling system services

```

[Insert print simulation result here]

```
andrewss@andrewss-ThinkPad-T480: ~/CPSC_240/Assignmen...
andrewss@andrewss-ThinkPad-T480:~$ cd CPSC_240/Assignments/assignment6/print_programs
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Assignments/assignment6/print_program
s$ yasm -g dwarf2 -f elf64 print.asm -l print.lst
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Assignments/assignment6/print_program
s$ ld -g -o print print.o
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Assignments/assignment6/print_program
s$ ./print
1 + 2 + 3 + ... + 100 = 5050
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Assignments/assignment6/print_program
s$
```

```
(gdb) x/uh %sum
0x402018: 5050
(gdb) x/xh %sum
0x402018: 0x13ba
(gdb) x/5ub %ascii
0x40201a: 53 48 53 48 10
(gdb) x/5xb %ascii
0x40201a: 0x35 0x30 0x35 0x30 0x0a
(gdb) x/5cb %ascii
0x40201a: 53 '5' 48 '0' 53 '5' 48 '0' 10 '\n'
(gdb) |
```

Registers		
rax	0x5	5
rbx	0x1	1
rcx	0x4010b4	4198580
rdx	0x5	5
rsi	0x40201a	4202522
rdi	0x1	1
rbp	0x0	0x0
rsp	0x7fffffff060	0x7fffffff060
r8	0x0	0
r9	0x0	0
r10	0x0	0
r11	0x206	518
r12	0x0	0

Integer registers

All registers

Close

Help

[Insert print simulation result analysis here]

sum = 5050

ascii = "0000" = 30h 30h 30h 30h

first digit

$\overset{ax}{50} \overset{bx}{50} = \overset{ax}{5} R \overset{dx}{50}$

add byte [ascii + 0], al = 5 + 30h = 35h = '5'
ascii = "5000", 10

second digit

$\overset{ax}{50} \overset{bx}{100} = \overset{ax}{0} R \overset{dx}{50}$

add byte [ascii + 1], al = 0 + 30h = 30h = '0'
ascii = "5000", 10

third digit

$\overset{ax}{50} \overset{bx}{10} = \overset{ax}{5} R \overset{dx}{0}$

add byte [ascii + 2], al = 5 + 30h = 35h = '5'
ascii = "5050", 10

fourth digit

$\overset{ax}{0} \overset{bx}{1} = \overset{ax}{0} R \overset{dx}{1}$

add byte [ascii + 3], al = 0 + 30h = 30h = '0'
ascii = "5050", 10

ascii = "5050" = 35h 30h 35h 30h