



WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

99:749389476 ERS

https://tutorcs.com

SEC204

Overview



- Introduction
- Integer arithmetic

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476



WeChat: cstutorcs

INTRODUCTION Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

NUMERIC DATA

- The core numeric figure 14.3 for the IA32 platform are:
 - Unsigned integer
 - Signed integers
 - Binary-coded decima Chat: cstutorcs

 - Packed binary-coded decimal
 Single-precision floating point

 Project Exam Help
 - Double-precision floating point tutores@163.com
 Double-extended floating point

QQ: 749389476

STANDARD IN

• The basic IA-32 plants 4 integer sizes:

• Byte: 8 bits

• Word: 16 bits

• Doubleword: 32 byseChat: cstutorcs

• Quadword: 64 bits Assignment Project Exam Help

- What is the range phunsigned integers you can represent with a word?
- What is the range Quigned integers (two's complement) you can represent with a word? https://tutorcs.com



WeChat: cstutorcs

INTEGER ARITHMETIC Email: tutores@163.com

QQ: 749389476

ADDITION

- add source, desting
 - Adds source to destination is placed in destination
- Can define the size of data element to be moved
 - addl: 1 for 32-bit Wood hat acstutores
 - addw: w for 16-bit word value
 - addb: b for 8-bit Assignment Project Exam Help
- Create an assembly program addnum or that performs the following functionality

```
int main() {
  int data = -40;
                             O: 749389476
  int b = 0;
  data = data +(-10)+(-200)+80+210
return 0;
                                   tutorcs.com/
```

In command line:

```
$ as —o addnum.o addnum.s
$ ld —o addnum addnum.o
 ./addnum
$ echo $?
```

SUBTRACTION

- sub source, destination source). The result of the subtraction is placed in cestination.
- Can define the size of datalentement: to sturrowed
 - subl:1 for 32-bit long word value
 - subw: w for 16-bit Avssignanent Project Exam Help
 - subb: b for 8-bit byte value
- Create an assembly programsubnuttottostostoche following functionality

In command line:

```
$ as -o subnum.o subnum.s
$ ld -o subnum subnum.o
$ ./subnum
$ echo $?
```

INCREMENTING CAREMENTING

- inc destination
 Increases destination by
- dec destination

 Decreases destination by 1

 WeChat: cstutorcs
- Create an assembly program count50.s that performs the following functionality:
 Assignment Project Exam Help

```
int main() {
    int data = 50;
    int b = 20;
    for (b=20; b<data; b=b+1) {
        printf("value of b: %nO:b749389476
    }
return 0;
}

https://tutorcs.com</pre>
```

In command line:

```
Email: tutorcs@163. Comp-o count50.o count50.s $ ld -dynamic-linker /lib/ld-linux.so.2 -lc -o count50 count50.o $ ./count50.o $ ./count50 $ echo $?
```

MULTIPLICATIN

- mul source For unsigned integers. The source.
 - integers. The transfer of integers. The complete integers in the size of
- mov \$5, %eax mull \$10
 movl %eax, result Result has value of 50.
 WeChat: cstutorcs
 MeChat: cstutorcs</li
- Assemble, link and run plegnami multeste from BEcom

QQ: 749389476

MULTIPLICATIN

```
UL instruction
# multest.s - An example of
.section .data
data1:
   .int 315814
data2:
   .int 165432
                        WeChat: cstutorcs
#quad is 64-bits
result:
   .quad 0
                        Assignment Project Exam Help
output:
   .asciz "The result is %qd\n"
.section .text
                        Email: tutorcs@163.com
.globl start
_start:
  nop
                         QQ: 749389476
  movl data1, %eax
  mull data2
  movl %eax, result
                        https://tutorcs.com
  movl %edx, result+4
  pushl %edx
```

Cont:

```
pushl %eax
  pushl $output
  call printf
  add $12, %esp
  pushl $0
  call exit
```

MULTIPLICATIN

- imul source For signed integers. The source.
- is implied (DX:AX) and is double the size of
- imul source, destination cstutores. For signed integers. The destination must be a register.
- imul multiplier, Assignment Project Exam Help For signed integers. Multiplier is a value, source can be a register or value in memory, destination must be a register. Multiplier * source = destination Email: tutorcs & 163.com

QQ: 749389476

DIVISION

• div divisor

For unsigned integers. The little is implied and must be stored in the AX register (if 16-bits), the Data tegrs (if 32-bits), or the EDX:EAX (if 64-bits).

Dividend	WeChat: Divident Size	CSTUTORCS Quotient	Remainder
AX	16bits ssignm	ent Project	t Evam Heln
DX:AX	32bits	AX TOJECT	t Exam Help
EDX:EAX	^{64bi} Email: tu	utorcs@16	3 COM

QQ: 749389476

DIVIDING

```
# divtest.s - An example of
                                       truction
.section .data
dividend:
   .quad 8335
divisor:
   .int. 25
                          WeChat: cstutorcs
quotient:
   .int. 0
                          Assignment Project Exam Help $12, *esp
remainder:
   .int 0
output:
   asciz "The quotient is &d, and the remainder is &d\n" | 500 | 163.COM
.section .text
.globl start
_start:
                          QQ: 749389476
  nop
  movl dividend, %eax
  movl dividend+4, %edx
                          https://tutorcs.com
  divl divisor
```

Cont:

```
mov1 %eax, quotient
movl %edx, remainder
pushl remainder
pushl quotient
pushl $output
call printf
call exit
```

SIGNED DIVISION

• idiv divisor

For signed integers. The integers in the AX register (if 16-bits), the DX:AX register bits), or the EDX:EAX (if 64-bits).

Dividend	WeChat: cstutor Divident Size Quotient	CS Remainder
AX	16bits ssignment Pro-	iect Evam Heln
DX:AX	Assignment Projection Ax	DX DX
EDX:EAX	^{64bi} Email: tutores@	163 com

QQ: 749389476

MULTIPLYING, DESCRIPTING

• sal destination sal shifter, destination sal shifter, destinated specifies the number of bits to shift. In binary, sal multiplies by 2.

WeChat: cstutorcs

• sar destination Assignment Project Exam Help sar shifter, destination

Shift arithmetic right. If used, shifter specifies the number of bits to shift. In binary, sar divides by 2.

QQ: 749389476

LOGICAL OPER LOGICA OPER LOGICA

• AND, OR, XOR xor source, desti

Performs logical XOR funding the result. Same format for AND, OR.

WeChat: cstutorcs

• not destination Assignment Project Exam Help Performs a NOT instruction. Each bit of destination is inverted.

Email: tutorcs@163.com

QQ: 749389476

FURTHER REA 🗎 🗟

• Professional Assemblification, chapters 7-8

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476