به نـــام خدا - امتحان میان‌ترم سوم ساختار و زبان کامپیوتر - دانشگاه صنعتی شریف - دی ماه ۱۴۰۲ - وقت: ۱۰۰ دقیقه

نام دانشجو: شماره دانشجویی: نمره از ۲۰:

۱- در جدول زیر، به ازای هر مجموعه از دستورات، یک یا دو دستور (با توجه به تعداد جای خالی مشخص شده در ستون‌های زوج) که عملیات معادل آن مجموعه را انجام می‌دهد بنویسید. (۳.۵ نمره)

Sample Code	Equivalent Instruction	Sample Code	Equivalent Instruction
neg ax dec ax	---------- not ax (۰.۵ نمره)	push dx xor dx, dx mov dl, al shl dx, 11 shr ax, 5 or ax, dx pop dx	---------- ror ax, 5 (۰.۵ نمره)

cmp ax, 0
jge L1
mov dx, -1
jmp L2
L1:
mov dx, 0
L2:
	---------- cwd
	(۰.۵ نمره)
		push dx
		mov dx, 0x8000
		and dx, ax
		shr ax, 1
		or ax, dx
		pop dx
			---------- sar ax, 1
			(۰.۵ نمره)

push bx
mov bx, 1
L1:
xor ax, bx
test ax, bx
jnz L2
shl bx, 1
jnc L1
L2:
pop bx
	---------- inc ax
	(۰.۵ نمره)
		push ax
		cmp cx, 0
		jz L2
		L1:
		mov ax, word[di]
		cmp ax, word[si]
		jz L2
		add di, 2
		add si, 2
		loop L1
		L2:
		pop ax
			---------- cld
			(۰.۵ نمره)
			---------- repnz cmpsw
			(۰.۵ نمره)

۲- فرض کنید پیش از اجرای این دستور مقادیر ثبات‌های st0 و st1 به ترتیب 0.1 و 2 می‌باشد. با استفاده از دو دستور FPU مقادیر این ثبات‌ها را به 1 و 20 (به همین ترتیب) تغییر دهید به صورتی که تغییری در مقدار سایر ثبات‌های st ایجاد نشود. (۱ نمره)

```
----------   divp st1    (۰.۵ نمره)
----------   fld1        (۰.۵ نمره)
```

۳- با توجه به segment data تعریف شده، پس از اجرای دستور mov، مقدار ax چه خواهد بود؟ (۱ نمره)

```
segment .data
L:  dw 5, -2

segment .text
mov ax, [L + 1]
```

با در نظر داشتن little endian بودن x86 و اندازه 16 بیتی ثبات‌ها در این کد، مقدار ax برابر 0xFE00 خواهد بود.

FE (۰.۵ نمره)

00 (۰.۵ نمره)

۴- اگر مقدار bx پیش از اجرای کد زیر برابر 6 باشد، پس از اجرای کد مقدار ax چه خواهد بود؟ (۱.۵ نمره)

```
        call   func
        jmp    end_func

func:
        cmp    bx,   1
        je     if
else:
        push   bx
        dec    bx
        call   func
        pop    bx
        mul    bx
        ret
if:
        mov    ax,   1
        ret
end_func:
```

برنامه بالا پیاده‌سازی بازگشتی تابع فاکتوریل است و مقدار ax پس از اجرای آن برابر 720 خواهد بود.

در صورت ارائه جواب درست (۱.۵ نمره)
در صورت ارائه جواب نهایی غلط و ارائه نتیجه هر مرحله، (۰.۵ نمره) به ازای عملیات صحیح در هر مرحله داده شود.

۵- اگر مقدار ax و bx به ترتیب 3 و 12 و AF=0 باشد، پس از اجرای دستورات زیر ax چه خواهد بود؟ (۱ نمره)

```
aaa              ax = 3
add  ax,  bx     ax = 15
aam              ah = 1, al = 5 => ax = 261 (0x01 (۰.۵ نمره) 05(۰.۵ نمره))
```

۶- تابع زیر را به صورتی تکمیل کنید که یک عدد از ورودی خوانده، در صورت زوج بودن، عدد یک و در صورت فرد بودن، عدد صفر را در خروجی چاپ کند. قرار است این تابع در یک سیستم ۶۴ بیتی با سیستم عامل لینوکس توسط یک برنامه C فراخوانی شود. این تابع ورودی و خروجی ندارد. فرض کنید که تمامی موارد لازم دیگر برای اجرای درست این کد رعایت شده و rsp پیش از فراخوانی این تابع alignment صحیح داشته باشد. حالت آدرس‌دهی پیشفرض relative (ممکن است جاهای خالی از دستورات بیشتر باشد، در صورت بیشتر بودن، فضای خالی اضافه را پر نکنید). (۳ نمره)

```
default rel

segment   .data
sf:   db    "%ld",   0
pf:   db    "%ld",   10, 0

segment   .text
global   even
extern   printf
extern   scanf
even:
----------              sub    rsp,   8          (۰.۲۵ نمره)
----------              lea    rdi,   [sf]       (۰.۲۵ نمره)
----------              lea    rsi,   [rsp]      (۰.۲۵ نمره)
----------              mov    al,    1          (۰.۲۵ نمره)
call scanf wrt ..plt
----------              lea    rdi,   [pf]       (۰.۲۵ نمره)
----------              mov    rsi,   [rsp]      (۰.۲۵ نمره)
----------                                       (۰.۵ نمره)
and   rsi,   1
----------              xor    rsi,   1          (۰.۲۵ نمره)
----------              mov    al,    1          (۰.۲۵ نمره)
call printf wrt ..plt
----------              add    rsp,   8          (۰.۲۵ نمره)
----------                                       (۰.۲۵ نمره)
ret
```

۷- تابع mod را به صورتی کامل کنید که

۱. ورودی‌ها به ترتیب در di و si به تابع داده شوند.
۲. خروجی در ax باشد.
۳. مقدار رجیسترهای bp و bx پس از بازگشت با تابع برابر با مقدار آن‌ها قبل از فراخوانی تابع باشد.
٤. این کار را با اضافه کردن کمترین دستور ممکن انجام دهید.

(۳ نمره)

```
mod:
----------              push   bx                          (۰.۵ نمره)
----------              xor   dx,   dx   OR   mov   dx,   0   (۰.۵ نمره)
----------              mov   ax,   di                     (۰.۵ نمره)
mov bx, ----------      si                                 (۰.۲۵ نمره)
div bx
----------              mov   ax,   dx                     (۰.۵ نمره)
----------              pop   bx                           (۰.۵ نمره)
----------                                                 (۰.۲۵ نمره)
ret
```

۸- برنامه C زیر، دو عدد از ورودی خوانده و عدد اول به توان عدد دوم را در خروجی چاپ می‌کند. پارامترهای بلوک asm را کامل کنید (برنامه با intel syntax کامپایل می‌شود). (۱.۵ نمره)

```
#include <stdio.h>
```

```
int main() {
   unsigned long int base, power, result;
   scanf("%ld %ld", &base, &power);
   asm volatile("mov rax, 1;"
                "cmp rcx, 0;"
                "jle end_L;"
                "L:"
                "    mul rbx;"
                "    loop L;"
                "end_L:"
                : ----------          "=a"(result)              (۰.۵ نمره)
                : ----------          "b"(base), "c"(power)     (۰.۵ نمره)
                : ---------- );         "rdx"                   (۰.۵ نمره)
   printf("%ld\n", result);
   return 0;
}
```

۹- کد ماشین دستورات زیر را محاسبه کنید. (آدرس منطقی برچسب arr که یک آرایه از نوع بایت است، برابر 002Ah و برچسب lbl برابر 09F6h می‌باشد) .(۴.۵ نمره)

add arr[si], 97	100000 0 0۰.۲۵ 01 000 100۰.۲۵ 00101010 01100001۰.۲۵
call far ptr ds:arr+6	11111111۰.۲۵ 00 010 110۰.۲۵ 00110000 ۰.۲۵ 00000000 ۰.۲۵
mov dx, -8	1100011 1۰.۲۵ 11 000 010۰.۲۵ 11111000 ۰.۲۵ 11111111۰.۲۵
lock sbb word ptr arr+16[bx-2], 64	11110000۰.۲۵100000 11۰.۲۵01 011 111 00111000 01000000۰.۲۵
repz cmpsb	11110011۰.۲۵ 10100110۰.۲۵
jmp lbl-8 ; current address is 1000h	11101001۰.۲۵ 1110 1110 ۱۱۱۱1001۰.۲۵

موفق و سربلند باشید – سربازی آزاد و اکبرزاده