

1291 / 9 / 11

الطفاً خوانا، مستدل و مفید با شیخ کفران را به هدیه . مدت : دو ساعت

५५.

۱- جدول زیر را تکمیل و در پاسخنامه خود وارد کنید :

| مقدار عدد دیگری نمایش دهد یا ری | نمایش مکمل ۲ | نمایش مکمل ۱۰ | نمایش مکمل ۱۴ |
|------------------------------------|--------------|---------------|---------------|
| 37.6 | | | |
| - 0.87 | | | |
| - 43.75 | | | |
| - 16.16 | | | |

توجه: تعداد ارقام
صحیح یا اعشاری
باید در حد لزوم
(نه بیشتر و نه کمتر)
باشد. این تعداد
ممکن است برای
رأی ۴ و ۵ متفاوت باشد.

۲-۱ نشان دهنده و اثبات کننده که اگر نمایش یک عدد علامت دار (بیشتر یا کمتر از ۲) به صورت

$$N = (a_{m-1}, a_{m-2}, \dots, a_1, a_0)$$

$a_{n-1} = 0 \rightarrow +$ بیت عدد
 $a_{n-1} = 1 \rightarrow -$

آن "۶۵ مقدار N از می گلب فرمول:

$$N = -a_{n-1} 2^{n-1} + \sum_{i=0}^{n-2} a_i 2^i$$

به راستی این

$N = (1011010)_2$ را به کد گری و با این کد به

۲-۲ مثال عددی :
حساب کنید.

۳ - محاسبه عدد $x = -85 + (36/92 - 49)/2$ را با ماشین :

$M_1 =$ stack machine

$M_2 =$ Accumulator based

$M_3 =$ 3-address machine

به زبان اسمبلی برنامه نویسی کنید. جهت انتقال برای ماشین M_2 دستور تفريق نداريم.

۴ - جدول زیر را کامل کرده ، مشخص کنید کدام نوع ماشین امروزه سرریز و محبوب تر است (چرا؟). [توضیح و معرفی کوتاه از m ، CPI و T_{cycle} به هم]

| | m | CPI | T_{cycle} |
|------|-----|-------|-------------|
| Risc | ? | ? | ? |
| Cisc | ? | ? | ? |

[در هر خانه جدول بنویسید کمتر یا بیشتر]

۳

۵ - جدول مربوط به مقایسه دو عدد A و B را برای انجام پرسش با ارائه یک توضیح منطقی و مختصر تکمیل نمایید.

| نوع اعداد | نام دستور پرسش | شرط پرسش | عبارت منطقی مبتنی بر بیت های وضعیت |
|------------|----------------|------------|------------------------------------|
| بدون علامت | BHI | $A \geq B$ | ? |
| بدون علامت | BLOE | $A \leq B$ | ? |
| با علامت | BGT | $A > B$ | ? |
| - | BE | $A = B$ | ? |

۲

۶ - یک ماشین دارای دستورات ۱۶ بیتی است. آدرس ۴ بیتی که می تواند برخی دستورات یک آدرس دارند و بقیه ۲ آدرس. اگر دستورات ۲ آدرس m تا باشد ، حداکثر دستورات یک آدرس چقدر است؟

۲

۷ - در میکروکنترلر ۸۰۵۱ دستورات دارای مقدار جابجایی (offset) بیش از ۸ بیت برای آدرس نیستند. آیا این بیان معنی است که این پردازنده نمی تواند آدرس ۹ بایت را در حافظه مورد رجوع قرار دهد؟ اگر جوابتان منفی است ، چه راهکاری اندیشیده شده است؟ (احس برزین)

۲

۸ - فرق آدرس دهی Auto Predecrement (یعنی پیش کاهش آدرس مورد استفاده به طور خاص کار و بنام دستور) را با کتیوه آدرس دهی دستور POP بیان کنید.

۲

۹ - زیر برنامه $mul10(int a, int b)$ را به اسمبلی بنویسید و کتیوه واخوانی آن را در اسمبلی بیان نمایید. مدقق باشید

$$mul10(int a, int b) \{$$

$$\quad x = a;$$

$$\quad \text{if } x > 0 \quad b = 10 * x \text{ else } b = x / 10;$$

$$\}$$

(نوع int ۳۲ بیتی است)

ص ۲

۳

۱۳۹۱/۹/۱۱

تصحیح امتحان میان‌ترم زبان کامپیوتر و اسمبلر

۴۰ - ۱۲۶

| مقدار | نمایش مکمل ۲ | نمایش مکمل ۱۰ | نمایش مکمل ۱۶ |
|--------|----------------|---------------|---------------|
| 37.6 | 0100101,1001 | 37.6 | 25.9 |
| -0.87 | 1,00100010 | 9.13 | F.22 |
| -43.75 | 1010100.01 | 56.25 | D4.4 |
| -16.16 | 101111.1101000 | 83.84 | EF.D8 |

۱ -

۴

توضیح: ۱- به ازای هر رقم دهدهی ۴ رقم باینری و یک رقم شش‌دهی لازم داریم.

۲- اگر $N = (a_{n-1}a_{n-2}...a_1a_0)_r$ باشد $N = (r^N - N)$ نمایش $N = -N$ بنا بر این مقدار

ارقام صحیح عدد، مقدار ارقام منفی آنرا تعیین می‌کند.

۳- صواب در تبدیل یک عدد از یک مبنا به مبنا دیگر، مقدار ارقام نامحدود و تکرار نشود
مثلاً $(37.6)_{10} = 0100101.10011001..._{2}$ ولی به بی‌نهایت مقدار ارقام یا بیت 3.3×10^4 \log_{10} تعداد ارقام اعشاری

یعنی ۴ بیت بسته می‌کنیم.

۴- برای نمایش و مکمل شش‌دهی، مقدار عدد مثبت را در مبنا ۱۶ می‌نویسیم و سپس نسبت به ۱۶ مکمل می‌گیریم:

$$0.87 = 0.1101111010...$$

$$0.87 = 0.D E \Rightarrow -0.87 = C_{16}(0.DE) = F.22$$

این عدد را می‌توان به دو طریق به دست آورد: روش اول همان فرمول بالا (۲) است:

$$C_{16}(0.DE) = (16^1 - 0.DE) = (F.22)$$

روش دوم این است که از حالت شروع می‌کنیم و اولین رقم را نسبت به ۱۶ و بقیه را نسبت به F مکمل می‌گیریم.

$$43.75 = (0101011.1100)_2 \Rightarrow -43.75 = (1010100.01)_2$$

$$= (2B.C)_{16} \Rightarrow -43.75 = (D4.4)_{16}$$

$$16.16 = 010000.001010001111 \Rightarrow -16.16 = (101111.1101000)_2 =$$

$$= (10.28)_H \Rightarrow -16.16 = FE.D8$$

$$N = (a_{n-1}, a_{n-2}, \dots, a_1, a_0)_2 \quad 1-2$$

اگر $a_n = 0$ ، فرمول $N = -a_{n-1} 2^{n-1} + \sum_{i=0}^{n-2} a_i 2^i = \sum_{i=0}^{n-2} a_i 2^i$ درست است.
اگر $a_n = 1$ ، پس چنین می کنیم:

$$-N = C_2(N) = C_1(N) + 1$$

$$-N = (1-a_{n-1}, 1-a_{n-2}, \dots, 1-a_1, 1-a_0) + 1$$

$$= \sum_{i=0}^{n-2} (1-a_i) 2^i + 1 = \sum_{i=0}^{n-2} 2^i - \sum_{i=0}^{n-2} a_i 2^i + 1$$

$$= 2^{n-1} - \sum_{i=0}^{n-2} a_i 2^i = 2^{n-1} - \sum_{i=0}^{n-2} a_i 2^i$$

و این همان فرمول مورد نظر است:

$$\Rightarrow -N = -2^{n-1} + \sum_{i=0}^{n-2} a_i 2^i$$

روش اول: $N = (1011010)_2 \Rightarrow -N = 0100110$
 $\downarrow \downarrow \downarrow \downarrow \downarrow \downarrow \downarrow$
 $2^5 2^4 2^3 2^2 2^1 2^0$
 $= 32 + 4 + 2 = 38 \Rightarrow N = -38$

روش دوم: $= -2^6 + (2^4 + 2^3 + 2^1) = -64 + 26 = -38$

$$X = -85 + (36/92 - 49)/2$$

M_1 : stack machine

Push 36
Push 92
Div
Push 49
SUB
Push 2
Div
Push 85
SUB
POP X

M_2 : Accumulator based

Ld Ai 36 (i: immediate addressing)
Div Ai 92
StA temp1
Ld Ai 49
CMPA $A \leftarrow \bar{A}$
INC A $A \leftarrow A+1 \equiv -A$
Add A, temp1
Div A, 2 $\equiv \text{shr } A$
StA temp2
Ld Ai 85
CMPA
INCA
Add A, temp2
StA X

M_3 : 3-address machine

oliv A, 36, 92
sub A, A, 49
shr A
sub A, A, 85
St X, A

۸- مقداردهی Auto Predecrement مقدار نیات آریسر را یک واحد (۱ بایت، ۲ تا بایت ۱۶ بیت و ۴ تا بایت ۳۲ بیت) کاهش می دهد، سپس کارمورد نظر را روی داده این به گونه نکات این نیات انجام می دهد.

۲ POP برعکس این کار را انجام می دهد یعنی اول داده را می خواند، سپس مقدار آن را برگرداند: SP را یک واحد افزایش می کند. در واقع POP مدل Post increment است و Push مدل آریسر دهی Auto predecrement.

برنامه اصلی

```
mov Ax, a
push ax
lea bx, b
push bx
call mul10
}
```

زیر برنامه

۹- mul 10:

```
mov BP, SP ; BP ← SP
add SP, -4 ; رزرو ۴ بایت
mov BX, 10 ; BX ← 10
mov AX, 4[BX] ; خواندن a
mov -4[BP], AX ; a ← a
CMP AX, 0
BLE L2 ; if a ≤ 0 goto L2
IntMul 10
mov BX, [BP]
mov [BX], AX ; b ← 10x
goto L3
L2: IntDiv 10 ; ax ← ax / 10
mov BX, [BP]
mov [BX], AX ; b ← x / 10
L3: add SP, 4
ret
```


ع -

| | n | CPI | T_{cycle} |
|------|------|------|-------------|
| RISC | بسته | بسته | کته |
| CISC | کته | بسته | کته |

$$T_{ex} = n \times CPI \times T_{cycle}$$

\uparrow \uparrow \uparrow
 تعداد دستورات برنامه تعداد کلاک طول زمان اجرای یک کلاک

امروزه $T_{ex_RISC} \leq T_{ex_CISC}$ زیرا وقتی از RISC استفاده می کنیم، معمولاً تعداد زیادی بایت داخل برنامه وجود دارد که مراجعه به حافظه را کاهش می دهد و عملیات آنرا درونی انجام می شوند و بنابراین به دلیل تنوع کمتر دستورات و مدلهای آدرس دهی، طول دستورات کمتر و اجرا آن ها سریعتر از دستورات CISC است و این باعث می شود که تعداد بسته دستورات برنامه RISC باز هم زودتر از برنامه مدل CISC اجرا شود.

| نوع اعداد | نام دستوری | شرطی | عبارت منطقی |
|------------|------------|------------|--|
| بدون علامت | BHI | $A > B$ | $C = 1$ زیرا $A - B > 0 \Rightarrow A + 2^n - B \geq 0 \Rightarrow 2^n + (A - B) \geq 0 \Rightarrow C = 1$ |
| بدون علامت | BLOE | $A \leq B$ | $C = 0$ یا $Z = 1$ |
| با علامت | BGT | $A > B$ | $\overline{SV} + SV = 1 \Leftrightarrow S = V$ $A - B > 0$ یعنی علامت مثبت و جواب بی سرریز باشد |
| - | BE | $A = B$ | $Z = 1$ و یا علامت منفی و جواب سرریز نباشد |

۶ - دستورات ۱۶ بیتی هستند. آدرس ۶ بیتی اند.
 پس دستورات ۲ آدرس ۱۲ بیتی بایت آدرس ۴ بیتی بایت که عملیات (opcode) دارند. لذا حد اکثر تعداد چنین دستوراتی $2^4 = 16$ است که طبق کدال فقط n تا تعریف شده است. بنابراین $16 - n$ که (۶ بیتی) باقی مانده است که ۱۶ دستورات (۱۶ آدرس) را تعریف کنند. چون ۶ بیتی از این دستورات ۱۶ آدرس به یک آدرس می گذارند (۱۶ آدرس) را تعریف کرده اند، ۶ بیتی باقیمانده ۲۷ حالت مختلف را تعریف می کنند \Rightarrow حد اکثر تعداد دستورات ۱۶ آدرس: $1.24 - 64n$ $(16 - n) \times 2^7 = 1.24 - 64n$

۷ - اینکه جایابی با ۸ بیت نمیشود راهی کد فانی دسترسی به آدرس ۴/۲۵۵ نیست. کافی است یک بایت ۱۶ بیتی (DPTR) فرض کنیم وجود دارد (که دارد) که بیت ۱۶ آدرس را در حافظه می کند و دستورات نیاز به مقدار offset جمع می شود. ۳