

گزارش تمرین ششم درس زیر پردازنده و زبان اسمبلی

در این تمرین همان طور که گفته شده بود قرار است یک ماشین حساب با زبان اسمبلی پیاده سازی کنیم. (هم بخش های امتیازی و هم بخش اصلی پیاده سازی شده اند)

برای خواندن ورودی از کاربر از وقفه ۰۱ استفاده شده است و برای نمایش خروجی هم از وقفه ی ۰۹ استفاده شده است.

روند کلی کار به این صورت است که مقدار خوانده شده به عنوان عدد اول در رجیستر BX و عدد دوم در رجیستر DX ذخیره می شود.

در صورتی که کاراکتر h پس از وارد شدن هر عدد بیاید به این معنی است که عدد وارد شده را باید در مبنای ۱۶ در نظر گرفت که برای این کار نیز ابتدا مقدار عدد ورودی را با ۹ مقایسه می کنیم در صورتی که این عدد از ۹ کوچکتر بود یا با ۹ مساوی بود نیازی به تغییر وجود ندارد؛ در غیر این صورت باید رقم سمت چپ عدد در ۱۶ ضرب شود (توان یک پایه که ۱۶ می باشد) و عدد سمت راست هم در یک (توان صفر پایه) که با توجه به اینکه ضرب در یک تغییری ایجاد نمی کند از ضرب آن صرف نظر می شود. در نهایت جمع دو عدد بالا مقدار نهایی عدد در مبنای ده است.

همچنین اپراتور ورودی نیز در CH ذخیره می شود و با استفاده از آن میتوانیم بفهمیم که الان در حال گرفتن عدد دوم هستیم (اپراتور ها به ترتیب به شکل زیر در CH ذخیره می شوند)

*	۱
+	۲
-	۳
/	۴

و با توجه اینکه همگی از صفر بزرگتر هستند پس از گرفتن اپراتور CH حتما بزرگتر از صفر می باشد و این به این معنی است که باید ورودی دوم را بگیریم.

پس از گرفتن ورودی ها نوبت به محاسبه ی خروجی بر اساس اپراتور مورد نظر می رسد.

تنها نکته ی این قسمت این است که اگر تقسیم بر صفر باشد ارور زیر چاپ می شود و برنامه به اتمام می رسد.

ERROR -> The denominator must be the opposite of zero

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
AX=000D BX=0404 CX=0400 DX=0000 SP=0040 BP=0000 SI=0000 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=011F NU UP EI PL NZ NA PO NC
0744:011F 80FA00 CMP DL,00
-
AX=000D BX=0404 CX=0400 DX=0000 SP=0040 BP=0000 SI=0000 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=0122 NU UP EI PL ZR NA PE NC
0744:0122 750B JNZ 012F
-
AX=000D BX=0404 CX=0400 DX=0000 SP=0040 BP=0000 SI=0000 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=0124 NU UP EI PL ZR NA PE NC
0744:0124 8D164000 LEA DX,[0040] DS:0040=5245
-
AX=000D BX=0404 CX=0400 DX=0040 SP=0040 BP=0000 SI=0000 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=0128 NU UP EI PL ZR NA PE NC
0744:0128 B409 MOV AH,09
-
AX=090D BX=0404 CX=0400 DX=0040 SP=0040 BP=0000 SI=0000 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=012A NU UP EI PL ZR NA PE NC
0744:012A CD21 INT 21
-
ERROR -> The denominator must be the opposite of zero AX=090D BX=0404 CX=0400 DX=0040 SP=0040 BP=0000 SI=0000 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=012C NU UP EI PL ZR NA PE NC
0744:012C E99B01 JMP 02CA
-
```

4/0

در نهایت هم بخش مربوط به ساخت استرینگ خروجی می رسد.

در صورتی که پیش از این حداقل یکی از اعداد به صورت هگز وارد شده باشد باید خروجی هگز هم داشته باشیم که این مورد با متغیر IS_HEX بررسی می شود.

برای محاسبه ی خروجی به صورت دسیمال خروجی که از قبل محاسبه شده بود را تا زمانی که صفر شود بر ۱۰ تقسیم می کنیم که به ترتیب یکان دهگان صدگان و هزارگان عدد خروجی را می دهد؛ سپس این مقدار که در رجیستر BX ذخیره شده را داخل استرینگ می ریزیم (متغیر DEC_STR)

روشی که برای انتقال به استرینگ در نظر گرفته ایم به این صورت است که ابتدا نیمه ی بالایی رجیستر (BH) را که با ارزش ترین رقم خروجی نیز هست را بررسی می کنیم و در صورتی که صفر نبود به استرینگ منتقل می کنیم. همین روال برای سایر رقم های خروجی انجام می شود. (دو رقم پایین در BL هستند). فقط رقم سمت چپ است که برای آن صفر بودن بررسی نمی شود زیرا در صورتی که صفر هم باشد باید خروجی صفر را داشته باشیم.

برای تبدیل به هگز هم روال گفته شده برای دسیمال انجام می شود با این تفاوت که در این جا به جای ۱۰ باید ۱۶ باشد.

در تصاویر زیر نمونه از ورودی و خروجی محاسبه شده توسط برنامه را می بینیم:

```

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0744:01EC E9D200 JMP 02C1
-
AX=0000 BX=0064 CX=0136 DX=0060 SP=0040 BP=0000 SI=000F DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C1 NU UP EI PL ZR NA PE NC
0744:02C1 BA0C00 MOV DX,000C
-
AX=0000 BX=0064 CX=0136 DX=000C SP=0040 BP=0000 SI=000F DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C4 NU UP EI PL ZR NA PE NC
0744:02C4 B409 MOV AH,09
-
AX=0900 BX=0064 CX=0136 DX=000C SP=0040 BP=0000 SI=000F DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C6 NU UP EI PL ZR NA PE NC
0744:02C6 CD21 INT 21
-
-64 AX=0900 BX=0002 CX=0136 DX=000C SP=0040 BP=0000 SI=000F DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C8 NU UP EI PL ZR NA PE NC
0744:02C8 B409 MOV AH,4C
-
AX=4C00 BX=0064 CX=0136 DX=000C SP=0040 BP=0000 SI=000F DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02CA NU UP EI PL ZR NA PE NC
0744:02CA CD21 INT 21
-
Program terminated normally (0000)
-

```

```

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0744:01EE E9D200 JMP 02C3
-
AX=0000 BX=0002 CX=0100 DX=0002 SP=0040 BP=0000 SI=0010 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C3 NU UP EI PL ZR NA PE NC
0744:02C3 BA0E00 MOV DX,000E
-
AX=0000 BX=0002 CX=0100 DX=000E SP=0040 BP=0000 SI=0010 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C6 NU UP EI PL ZR NA PE NC
0744:02C6 B409 MOV AH,09
-
AX=0900 BX=0002 CX=0100 DX=000E SP=0040 BP=0000 SI=0010 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C8 NU UP EI PL ZR NA PE NC
0744:02C8 CD21 INT 21
-
-21 AX=0900 BX=0002 CX=0100 DX=000E SP=0040 BP=0000 SI=0010 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02CA NU UP EI PL ZR NA PE NC
0744:02CA B44C MOV AH,4C
-
AX=4C00 BX=0002 CX=0100 DX=000E SP=0040 BP=0000 SI=0010 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02CC NU UP EI PL ZR NA PE NC
0744:02CC CD21 INT 21
-
Program terminated normally (0000)
-

```

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0744:02C1 CD21          INT     21
12h AX=0968 BX=0012 CX=0031 DX=0022 SP=0040 BP=0000 SI=0025 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C3 NU UP EI PL NZ NA PO CY
0744:02C3 BA0E00      MOV     DX,000E
-
AX=0968 BX=0012 CX=0031 DX=000E SP=0040 BP=0000 SI=0025 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C6 NU UP EI PL NZ NA PO CY
0744:02C6 B409      MOV     AH,09
-
AX=0968 BX=0012 CX=0031 DX=000E SP=0040 BP=0000 SI=0025 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02C8 NU UP EI PL NZ NA PO CY
0744:02C8 CD21
18h AX=0968 BX=0012 CX=0031 DX=000E SP=0040 BP=0000 SI=0025 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02CA NU UP EI PL NZ NA PO CY
0744:02CA B44C      MOV     AH,4C
-
AX=4C68 BX=0012 CX=0031 DX=000E SP=0040 BP=0000 SI=0025 DI=0000
DS=0770 ES=0734 SS=0778 CS=0744 IP=02CC NU UP EI PL NZ NA PO CY
0744:02CC CD21      INT     21
-
Program terminated normally (0068)
```

بخش اعداد منفی :

برای این بخش علامت های ورودی ها را در متغیرهایی ذخیره می کنیم و به این ترتیب در زمان محاسبه ی نتیجه ی نهایی با بررسی علامت ها نتیجه را محاسبه می کنیم و در خروجی قرار می دهیم.

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
AX=0000 BX=0015 CX=0131 DX=0010 SP=0040 BP=0000 SI=0009 DI=0000
DS=0781 ES=0734 SS=0789 CS=0744 IP=02F6 NU UP EI PL ZR NA PE NC
0744:02F6 E9D200 JMP 03CB
AX=0000 BX=0015 CX=0131 DX=0010 SP=0040 BP=0000 SI=0009 DI=0000
DS=0781 ES=0734 SS=0789 CS=0744 IP=03CB NU UP EI PL ZR NA PE NC
0744:03CB BA0600 MOV DX,0006
AX=0000 BX=0015 CX=0131 DX=0006 SP=0040 BP=0000 SI=0009 DI=0000
DS=0781 ES=0734 SS=0789 CS=0744 IP=03CE NU UP EI PL ZR NA PE NC
0744:03CE B409 MOV AH,09
AX=0900 BX=0015 CX=0131 DX=0006 SP=0040 BP=0000 SI=0009 DI=0000
DS=0781 ES=0734 SS=0789 CS=0744 IP=03D0 NU UP EI PL ZR NA PE NC
0744:03D0 CD21 INT 21
-15AH=0900 BX=0015 CX=0131 DX=0006 SP=0040 BP=0000 SI=0009 DI=0000
DS=0781 ES=0734 SS=0789 CS=0744 IP=03D2 NU UP EI PL ZR NA PE NC
0744:03D2 B44C MOV AH,4C
AX=4C00 BX=0015 CX=0131 DX=0006 SP=0040 BP=0000 SI=0009 DI=0000
DS=0781 ES=0734 SS=0789 CS=0744 IP=03D4 NU UP EI PL ZR NA PE NC
0744:03D4 CD21 INT 21
```