

به نام خدا



درس ساختار و زبان کامپیوتر
نیم سال اول ۰۴-۰۳
استاد: دکتر اسدی

دانشکده مهندسی کامپیوتر

تمرین سری هشتم

- پرسش‌های خود را در سامانه CW و تالار مربوط به تمرین مطرح نمایید.
- پاسخ سوالات را تایپ نمایید.
- اسکرین‌شات‌ها، عکس‌ها، فایل‌های مربوط به سوال عملی، گزارش تمرینات عملی و PDF قسمت تئوری را در پوشه با نام به فرمت HWNUM_StudentID1_StudentID2 ذخیره نمایید. سپس آن را zip نمایید و در صفحه درس بارگذاری نمایید. به عنوان مثال یک فایل بارگذاری شده قابل قبول باید دارای فرمت HW1_400123456_403123456.zip باشد.
- هر دانشجو می‌تواند حداکثر دو تمرین را با دو روز تأخیر بدون کاهش نمره ارسال نماید.
- تمرینات عملی به صورت گروه‌های دو نفر تحویل داده شود.
- هر دو عضو گروه موظف هستند تمرینات خود را بارگذاری کنند.
- عواقب عدم تطابق بین پاسخ دو عضو گروه برعهده خودشان است.
- تحویل تمرین به صورت انگلیسی مجاز نیست. در صورت تحویل تمرین به صورت انگلیسی (حتی بخشی از تمرین) نمره تمرین موردنظر صفر در نظر گرفته می‌شود.
- در صورت مشاهده تقلب برای بار اول نمره هر دو طرف صفر می‌شود. در صورت تکرار نمره کل تمرینات صفر خواهد شد.
- استفاده از ابزارهایی مانند ChatGPT به منظور ابزار کمک آموزشی مجاز است به شرط آن که به خروجی آن اکتفا نشود.
- توجه شود که پروژه نهایی درس در گروه‌های چهار نفر تحویل گرفته می‌شود.
- سوالات با عنوان اختیاری نمره‌ای ندارند اما جواب دادن به آن‌ها کمک به سزایی در یادگیری درس می‌کند.

تمارین تئوری

۱. با در نظر گرفتن معماری‌های MIPS، 8086 و IBM، بررسی کنید که تفاوت‌های موجود در نمایش و پردازش اعداد floating-point در این سیستم‌ها چگونه بر دقت، سرعت و پیچیدگی پیاده‌سازی الگوریتم‌های عددی مانند matrix multiplication تأثیر می‌گذارد. انتظار می‌رود ابتدا مشخص کنید که هر معماری چگونه محاسبات floating-point را مدیریت می‌کند.

۲. در ISA مربوط به پردازنده‌های ARM، دستوری به منظور ضرب کردن مقدار موجود در یک رجیستر و یک مقدار عددی وجود ندارد. ابتدا به اختصار توضیح دهید دلیل این موضوع با توجه به cisc یا risc بودن پردازنده و همچنین کدگذاری^۱ دستور چگونه توجیه می‌شود. در ادامه روش جایگزینی به منظور ضرب کردن مقدار ثابت R4 در عدد ۱۹ و قرار دادن حاصل در ثابت R5 ارائه دهید. (مقدار نهایی R4 فاقد اهمیت است.)

۳. در یک روش نمایش اعداد اعشاری، اعداد به صورت یک آرایه ۳۲ بیتی نمایش داده می‌شوند که بیت اول برای علامت تخصیص داده شده است. بعد از آن، x بیت برای نمایش نما^۲ و y بیت برای بخش کسری^۳ استفاده می‌شود. این روش مشابه نمایش Single Precision Floating Point است، اما با مقادیر x، y و bias متفاوت. می‌دانیم نمایش عدد 7.7- در این سیستم معادل BBD99999 در مبنای 16 است. با استفاده از این اطلاعات، مقادیر x، y و bias را بیابید.

۴. (آ) با فرض این که نمایش‌های باینری سه عدد A و B و A+B به روش مکمل دو در n بیت جا می‌شوند، ثابت کنید اگر نمایش مکمل دوی A و B را با یک جمع‌کننده n بیتی با هم جمع کنیم، حاصل معادل نمایش مکمل دوی A+B می‌شود. تمام علامت‌های ممکن A و B را در نظر بگیرید. مثال:

```

1 n = 4
2 -3 -> 1101
3 -2 -> 1110
4 1101 + 1110 = 1011 -> -5
5
6 Hint: 1101 = 13 = 16 - 3

```

(ب) اگر نمایش‌های مکمل دو A و B در n بیت جا شوند ولی A+B نه، علامت‌های سه عدد A و B و حاصل جمع مکمل دو آن‌ها با یک جمع‌کننده n بیتی را بررسی کنید. مثلاً اگر در مثال بالا n=3 باشد.

تمارین عملی

۱. در این تمرین در ابتدا به شما یک عدد n داده می‌شود که تعداد اعداد ورودی می‌باشد. سپس در n خط بعدی، در هر خط یک عدد به شما داده می‌شود و شما باید بررسی کنید که جمع این اعداد از ابتدا تا کنون مربع کامل شده است یا خیر. تضمین می‌شود که جمع اعداد در بازه اعداد علامت‌دار ۶۴ بیتی جا شود. همچنین اعداد ورودی بزرگتر از صفر می‌باشند. نمونه ورودی و خروجی:

```
1 input:
2 5
3 1
4 3
5 4
6 1
7 9
8 output:
9 YES
10 YES
11 NO
12 YES
13 NO
```

در ابتدا عدد ۵ آمده یعنی ۵ ورودی داریم. سپس عدد ۱ آمده که مربع کامل است. در خط بعدی ۳ آمده و جمع اعداد ۴ شده که باز هم مربع کامل است. در خط بعدی ۴ را داریم که جمع اعداد تا کنون ۸ شده و مربع کامل نیست. سپس ۱ آمده است که با ۸ جمع شده است و نتیجه ۹ می‌شود که مربع کامل است و در نهایت نیز با ۹ جمع شده که نتیجه ۱۸ می‌شود که مربع کامل نیست.

۲. برنامه‌ای به زبان IBM s390x بنویسید که ورودی آن یک رشته است که این رشته فقط شامل حروف بزرگ و کوچک انگلیسی و اعداد می‌باشد. شما باید حاصل جمع اعداد موجود در رشته را خروجی دهید. در صورتی که عددی در رشته نبود، صفر را خروجی دهید. طول ورودی حداکثر ۱۰۰ کاراکتر است. نمونه ورودی و خروجی:

```
1 input:
2 IBMs390xx86assembly
3 output:
4 476
```

```
1 input:
2 Aa100bb1c
3 output:
4 101
```

۳. برنامه‌ای به زبان اسمبلی IBM s390x بنویسید که یک عدد دهدهی در ورودی بگیرد و اگر هم نمایش دهدهی آن هم نمایش دودویی آن پالیندروم بود، Yes را در خروجی چاپ کند در غیر این صورت No را در خروجی چاپ کند.

۴. برنامه‌ای به زبان IBM s390x بنویسید که عملکرد ضرب‌کننده ترتیبی^۴ را پیاده‌سازی کند. در محاسبه حاصل ضرب از کمترین تعداد جمع استفاده کنید. (برای این کار لازم است منطق بخش control test را به دقت پیاده‌سازی کنید.)

۵. برنامه‌ای به زبان اسمبلی IBM s390x بنویسید که یک رشته از اعداد ۰ و ۱ را ورودی بگیرد و با بررسی این رشته اندازه‌ی بزرگترین زیررشته‌ی ساخته شده از اعداد یک و اندازه‌ی بزرگترین زیررشته‌ی اعداد صفر را نمایش دهد. نمونه‌ی ورودی:

^۴shift and add multiplier

```
1 10000011011110001
```

نمونه‌ی خروجی:

```
1 4
```

```
2 5
```

۶. تابع `int sum(int* arr, int n)` را در اسمبلی IBM s390x با توجه به قرارداد صدا زدن (calling convention) پیاده کنید. این تابع همانطور که از اسمش پیداست، مسئول محاسبه کردن جمع اعضای آرایه `n` عضوی `arr` است که عضو اول آن `*arr` است.