



دستور کار:

- ❖ پس از ۱۰ روز تاخیر مجاز در مجموع کل تمرینات (کامپیوتری + دستی)، تحویل تمرین با تاخیر شامل جریمه می باشد (هر روز 25 درصد).
- ❖ نام فایل ارسالی را در قالب ACA_HW(number)_studentID بارگذاری شود.
- ❖ تمامی پروژه ها از لحاظ شباهت، کنترل و بررسی می شوند بنابراین از کپی کردن خودداری فرمایید چنانچه در صورت مشاهده صفر لحاظ خواهد شد و نیز در مجموع نمرات جریمه خواهید شد.
- ❖ پاسخ خود را در ادامه سوال و داخل فایل word قرار گرفته در سامانه قرار دهید و برای اسکن کردن پاسخ های خود از CamScanner استفاده کنید و طبق فرمت خواسته شده فایل را pdf شده اپلود کنید.
- ❖ راه ارتباطی با حل تمرین:

sara.zamani73@aut.ac.ir



1- کد زیر را در نظر بگیرید:

```
Loop: LD R1, 0(R2)
      DADDI R1, R1, #1
      SD 0(R2), R1
      DADDI R2, R2, #4
      DSUB R4, R3, R2
      BNEZ R4, Loop
```

(a) زمان این توالی دستورالعمل را برای pipeline 5 مرحله‌ای به همراه تعداد چرخه‌های مورد نیاز برای اجرای یک تکرار از حلقه بدون forward نشان دهید. فرض کنید که رجیسترها را می‌توان در یک چرخه، در حین writeback، نوشت و خواند. (تعداد چرخه‌های اجرای یک تکرار حلقه پس از مرحله EX دستور BNEZ به پایان می‌رسد)

(b) زمان این دنباله دستورالعمل را برای یک pipeline 5 مرحله‌ای به همراه تعداد چرخه‌های مورد نیاز برای اجرای یک تکرار حلقه همراه با forward نشان دهید.

(c) فرض کنید که مقدار اولیه $R3 = R2 + 396$ باشد.

(d) با فرض اینکه branch یک slot تاخیر دارد، کد را بازنویسی کنید تا آن slot پر شود.

2- افزایش عمق pipeline بر عملکرد^۱ از دو طریق چرخه clock سریع‌تر و افزایش stall به دلیل hazard داده و کنترل تأثیر می‌گذارد. فرض کنید ماشین اول یک pipeline 5 مرحله‌ای با 2 ns چرخه clock، ماشین دوم یک pipeline 12 مرحله‌ای با چرخه clock 1.2 ns است. pipeline 5 مرحله‌ای به دلیل hazard داده در هر پنج دستورالعمل دچار stall می‌شود، در حالی که pipeline 12 مرحله‌ای هر 8 دستورالعمل سه stall را تجربه می‌کند. علاوه بر این، branch ها 20٪ دستورالعمل‌ها را تشکیل می‌دهند و میزان پیش‌بینی نادرست برای هر دو ماشین 5٪ است.

(a) سرعت pipeline 12 مرحله‌ای نسبت به pipeline 5 مرحله‌ای با در نظر گرفتن فقط hazard داده چقدر است؟

^۱ performance



(b) اگر جریمه پیش‌بینی نادرست branch برای ماشین اول 2 چرخه باشد اما ماشین دوم 5 چرخه باشد، با در نظر گرفتن stalls ناشی از پیش‌بینی اشتباه branch، CPI هر کدام چقدر است؟

3- یک پیش‌بینی‌کننده همبسته^۲ (m,n) از رفتار m آخرین branch‌های اجرا شده برای انتخاب از بین 2^m پیش‌بینی‌کننده استفاده می‌کند که هر کدام یک پیش‌بینی‌کننده n-bit هستند. یک پیش‌بینی‌کننده محلی دو سطحی به روشی مشابه کار می‌کند، اما فقط رفتار گذشته هر branch را برای پیش‌بینی رفتار آینده بررسی می‌کند. یک trade-off طراحی با چنین پیش‌بینی‌کننده‌هایی وجود دارد: پیش‌بینی‌کننده‌های همبسته به حافظه کمی برای تاریخچه نیاز دارند، که به آن‌ها اجازه می‌دهد پیش‌بینی‌کننده‌های ۲ بیتی را برای تعداد زیادی از branch‌های جداگانه حفظ کنند (احتمال استفاده مجدد دستورالعمل‌های branch از همان پیش‌بینی‌کننده را کاهش می‌دهد)، در حالی که پیش‌بینی‌کننده‌های محلی به حافظه بیشتری برای نگهداری تاریخچه نیاز دارند و بنابراین محدود به ردیابی تعداد نسبتاً کمی از دستورالعمل‌های branch هستند. برای این تمرین، یک پیش‌بینی‌کننده همبسته (1،2) را در نظر بگیرید که می‌تواند چهار branch (نیاز به 16 بیت) را در مقابل یک پیش‌بینی‌کننده محلی (1،2) که می‌تواند دو branch را با استفاده از همان مقدار حافظه ردیابی کند. برای نتایج branch‌های زیر، هر پیش‌بینی، ورودی جدولی که برای پیش‌بینی استفاده می‌شود، هرگونه به‌روزرسانی در جدول، و نرخ پیش‌بینی نادرست نهایی هر پیش‌بینی‌کننده را ارائه دهید. فرض کنید تمام branch‌ها تا این مرحله گرفته شده است. هر پیش‌بینی‌کننده را با جدول زیر مقداردهی کنید:

² correlating branch predictor



Correlating predictor

Entry	Branch	Last outcome	Prediction
0	0	T	T with one misprediction
1	0	NT	NT
2	1	T	NT
3	1	NT	T
4	2	T	T
5	2	NT	T
6	3	T	NT with one misprediction
7	3	NT	NT

Local predictor

Entry	Branch	Last 2 outcomes (right is most recent)	Prediction
0	0	T,T	T with one misprediction
1	0	T,NT	NT
2	0	NT,T	NT
3	0	NT	T
4	1	T,T	T
5	1	T,NT	T with one misprediction
6	1	NT,T	NT
7	1	NT,NT	NT



Branch PC (word address)	Outcome
454	T
543	NT
777	NT
543	NT
777	NT
454	T
777	NT
454	T
543	T

- 4- فرض کنید ما یک پردازنده با pipeline عمیق داریم که برای آن یک branch-target buffer فقط برای branchهای شرطی پیاده‌سازی می‌کنیم. فرض کنید که جریمه پیش‌بینی اشتباه همیشه چهار چرخه است و جریمه خطای بافر همیشه سه چرخه است. نرخ hit 90 درصد، دقت 90 درصد و فرکانس انشعاب 15 درصد را در نظر بگیرید. پردازنده با branch-target buffer در مقابل پردازنده ای که جریمه branch دو چرخه ثابت دارد چقدر سریعتر است؟ یک چرخه clock پایه در هر دستورالعمل (CPI) را بدون branch stalls، یک فرض کنید.
- 5- یک خط لوله عمیق در نظر بگیرید که فقط یک BTB برای پرش های شرطی پیاده سازی کرده ایم. فرض کنید جریمه misprediction 5 سیکل و جریمه miss شدن در بافر همواره 4 سیکل است. نرخ hit را 90 درصد، accuracy 90 درصد و فرکانس پرش نیز 15 درصد است. سرعت یک پردازنده با داشتن BTB را با یک پردازنده ای که به طور ثابت 2 سیکل جریمه پرش دارد مقایسه کنید. هر دستور یک کلاک برای اجرا (بدون در نظر گرفتن stall پرش برای هر یک) نیاز دارد.
- 6- یک ریزمعماری با خط لوله 5 مرحله ای شامل fetch، decode، execute، memory، write-back همراه با دستورات زیر را در نظر بگیرید. تمام دستورات یک سایکلی هستند به جز LW و SW که 1+2 سایکلی و پرش ها 1+1 سایکلی اند. هیچ گونه forwarding وجود ندارد. فاز هر دستور در هر کلاک برای یک iteration از حلقه را نشان دهید.

(a) چه تعداد کلاک سایکل برای هر تکرار از حلقه به عنوان سربار پرش محسوب می شود؟



- (b) یک پیش بینی کننده پرش استاتیک در نظر بگیرید که از قابلیت backward branch در مرحله دیکد برخوردار است. حال چه تعداد کلاک سایکل در سربار پرش ها هدر می رود؟
- (c) یک پیش بینی کننده پرش داینامیک در نظر بگیرید. چه تعداد سایکل در یک پیش بینی صحیح هدر می رود؟

Loop:	LW	R3,0(R0)
	LW	R1,0(R3)
	ADDI	R1,R1,#1
	SUB	R4,R3,R2
	SW	R1,0(R3)
	BNZ	R4, Loop