

## Hodnocení

Za úplné a správné zodpovězení otázky je možné získat nejvýše počet bodů uvedený v hranatých závorkách. Rozsah odpovědi by měl pokrývat otázku samotnou, případně bezprostředně související problematiku, pokud je zřejmé, že odpověď by měla triviální charakter. Volba rozsahu odpovědi a její formulace velkou měrou vypovídá o porozumění tématu.

Celkem je možné získat 20 bodů, přičemž výsledné hodnocení je odvozeno z počtu získaných bodů. Znamka 1 je podmíněna získkem alespoň 17 bodů, známka 2 získkem alespoň 13 bodů a známka 3 získkem alespoň 10 bodů.

### Otázka 1 [1]

Vysvětlete, jak přidání dalších procesorů ovlivní dobu odezvy (*response time*) a propustnost (*throughput*) počítačového systému (např. webového serveru), který využívá více procesorů pro zpracování různých požadavků (obsluha konkrétního požadavku je sekvenční). Předpokládejte, že systém je málo zatížený a tedy žádné požadavky nečekají ve frontě.

### Otázka 2 [1]

Uvažujte následující procesory, které implementují stejnou instrukční sadu a vykonávají stejný program:

| Procesor | Frekvence | CPI |
|----------|-----------|-----|
| P1       | 2 GHz     | 1.5 |
| P2       | 1.5 GHz   | 1.0 |
| P3       | 3 GHz     | 2.5 |

Snažíme se snížit dobu vykonávání programu procesorem P1 o 20%, což má bohužel za následek zvýšení CPI o 20%. Na jaké frekvenci musí tento procesor pracovat, abychom toho dosáhli?

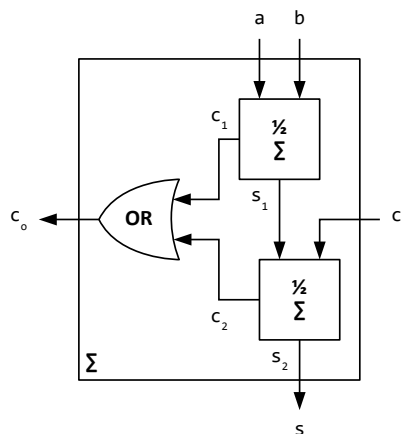
### Otázka 3 [1]

Simulační program tráví 10% času činností, kterou nelze paralelizovat. Pro zrychlení programu se používá výpočetní cluster se 16 uzly. Jakého zrychlení (oproti původnímu clusteru) bychom dosáhli spuštěním na clusteru se 32 uzly? Jaké je maximální (teoretické) zrychlení pro tento program?

### Otázka 4 [1]

Vysvětlete, jak byste převedli řadič jednocyklové datové cesty realizovaný pomocí kombinačního obvodu na realizaci pomocí ROM. Proč je takový převod možný a v jaké situaci by byl vhodný?

### Otázka 5 [2]



Odvoďte pravdivostní tabulku pro vstupy a výstupy úplné jednobitové sčítačky znázorněné na obrázku. Vysvětlete, jak se úplná sčítačka liší od poloviční sčítačky a pomocí hradel (nebo logických funkcí) funkcí vyjádřete vnitřní strukturu poloviční sčítačky.

### Otázka 6 [2]

Pro jednocyklovou datovou cestu na Obr. 1 (strana 2) napište ohodnocení řídicích signálů tak, aby datová cesta vykonala instrukci

`lw Rt, Offset (Rs)`

s následující interpretací:

$\text{Reg [Rt]} = \text{Mem [Reg [Rs] + \text{SignExt (Offset)}]$

Popište co datová cesta v průběhu vykonávání instrukce dělá a v kontextu toho zdůvodněte ohodnocení jednotlivých řídicích signálů.

### Otázka 7 [1]

Předpokládejte, že jednotlivé stupně pipeline (resp. odpovídající části jednocyklové datové cesty) mají následující latence:

| IF     | ID     | EX     | MEM    | WB     |
|--------|--------|--------|--------|--------|
| 200 ps | 150 ps | 120 ps | 190 ps | 140 ps |

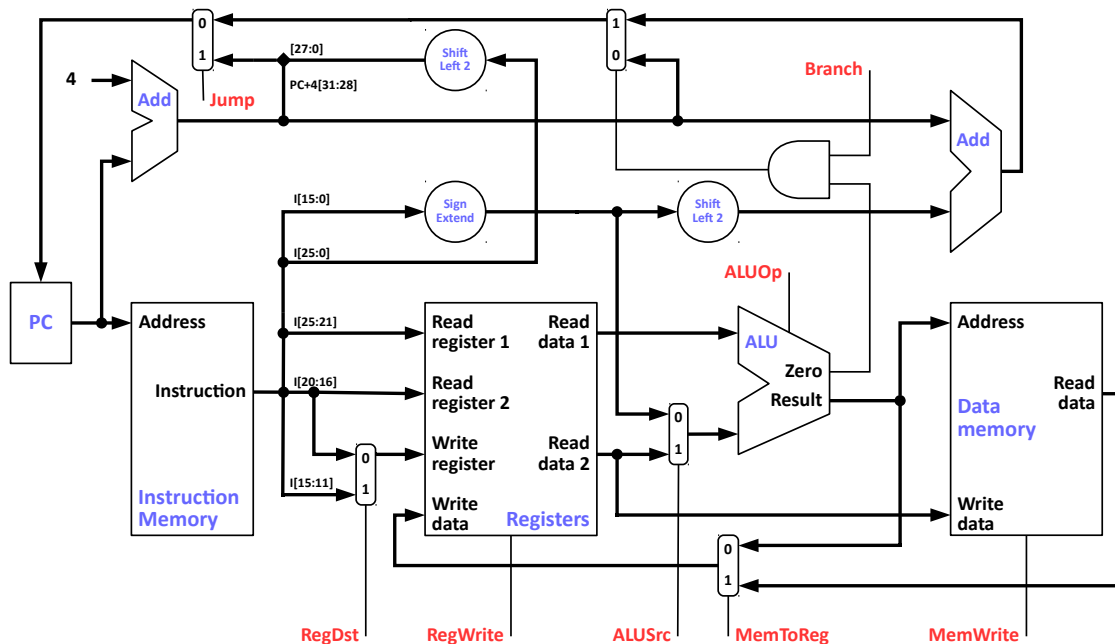
Jaká bude celková *latence* instrukce **lw** (**load word**) v případě jednocyklové a v případě zřetěžené (*pipelined*) datové cesty?

### Otázka 8 [1]

Jaké je vytížení zápisového portu registrového pole (% cyklů, kdy je používán) ve zřetěžené datové cestě (klasická 5-stupňová single-issue MIPS pipeline), pokud procesor vykonává program s následujícím instrukčním mixem:

| ALU | beq | lw  | sw  |
|-----|-----|-----|-----|
| 30% | 25% | 30% | 15% |

Předpokládejte, že při běhu programu nenastávají žádné hazardy, které by si vynucovaly zpoždování pipeline.



Obrázek 1

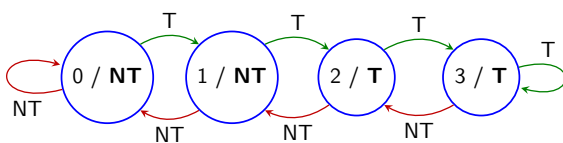
## Otázka 9 [2]

Vysvětlete jaký problém řeší a jak funguje forwarding/bypassing v datové cestě se zřetězeným zpracováním instrukcí (klasická 5-stupňová single-issue MIPS pipeline). Uveďte příklad krátké sekvence instrukcí, kde se dá forwarding/bypassing aplikovat.

## Otázka 10 [1]

Jaká je přesnost predikce 2-bitového dynamického prediktora na Obr. 2 (strana 2) pro následující posloupnost výsledků (stejně) instrukce podmíněného skoku (stav 0 uvažujte jako počáteční) :

T, T, T, NT, NT



Obrázek 2

## Otázka 11 [1]

Zpoždění pipeline v důsledku špatně předpověděných výsledků podmíněných skoků zvyšuje CPI. Jakého zrychlení by se dalo dosáhnout s 2-bitovým prediktorem, pokud bychom nahradili polovinu instrukcí větvení za speciální instrukci vykonávanou pomocí ALU? Předpokládejte, že správně i špatně predikované instrukce větvení jsou nahrazovány se stejnou pravděpodobností. Uvažujte následující instrukční mix a přesnost predikce 95%.

| registrové | beq | jmp | lw  | sw  |
|------------|-----|-----|-----|-----|
| 30%        | 10% | 5%  | 35% | 20% |

Uvažujte klasickou 5-stupňovou pipeline s fázemi IF-ID-EX-MA-WB a vyhodnocením podmíněného skoku ve fázi EX. Předpokládejte, že v programu nejsou žádné datové hazardy, a že se nevyužívají branch delay sloty.

## Otázka 12 [1]

Jsou data v tzv. sekundární paměti (pevný disk, SSD) přímo použitelná (mohou být načtena do registru nebo použita jako operand aritmetických operací) procesorem? Odpověď zdůvodněte.

## Otázka 13 [1]

Vysvětlete jak a proč stupeň asociativity ovlivňuje výkonost cache (předpokládejte, že ostatní parametry cache se nemění).

## Otázka 14 [2]

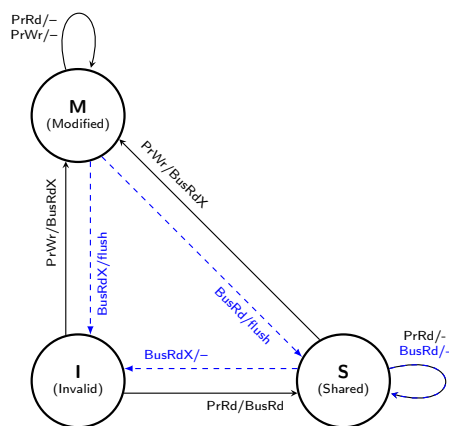
Pro 16 souvislých bloků paměti schematicky znázorněte mapování do 2-cestné množinově asociativní cache s 8 položkami (*cache lines*), tj. kde v cache by mohly být bloky uloženy. Znázorněte jak a k čemu se při přístupu do cache používají různé části adresy  $A$  v závislosti na počtu *cache lines* a jejich velikosti  $B$  v bajtech. Co všechno musí cache obsahovat a jaká by měla 8-položková cache režii, pokud by adresy byly 16-bitové a velikost cache line 4 bajty?

## Otázka 15 [2]

Přístupy do paměti programu běžícího na více procesorech způsobily následující posloupnost operací nad jednou konkrétní cache line:

1. **P2** write
2. **P0** read
3. **P1** read
4. **P0** write
5. **P2** read

Předpokládejte, že víceprocesorový systém používá write-back cache a koherenční protokol MSI, jehož přechodový diagram je znázorněn na obrázku 3. Pro každou operaci uveďte, jak budou reagovat a vzájemně komunikovat řadiče cache jednotlivých procesorů a jak se bude měnit stav cache line na jednotlivých procesorech.



Obrázek 3