

Hodnoceni

Uloha 1 [1]: 1b

- Pro stejnou zatez by se nic nezmenilo
 - Ani aktualni propustnost (odpovida zatezi, protoze pozadavky necekaji)
- Stoupnula by maximalni propustnost systemu

Uloha 2 [1]: 1b

Uloha 3 [1]: 1b

Uloha 4 [1]: 1b

Uloha 5 [2]: 2b

- Tabulku doporucuji psat tak, aby vstupy i vystupy tvorily binarni cislo s tim, ze c_i ma nejvyssi vahu u “vstupniho” cisla a c_o ma nejvyssi vahu u “vystupniho” cisla (tam to tak mate).

Uloha 6 [2]: 2b

Uloha 7 [1]: 0.5b

- V pripade pipeline musi byt hodinovy cyklus 200ps, protoze to je latence nejpomalejsi casti zpracovani instrukce, takže celkova latence (doba zpracovani instrukce) pri pruchodu pipeline je $5 \times 200\text{ps}$.

Uloha 8 [1]: 1b

Uloha 9 [2]: 2b

Uloha 10 [1]: 0-0.5b

- Uloha se ptala na presnost pri jednom pruchodu uvedenou sekvenci s pocatecnim stavem 0, tj. predikci NT.
- Druhy skok bude stale predikován NT, tedy spatne a prediktor spravne predpovi pouze 1 z 5 skoku.

Uloha 11 [1]: 0.5

- Uvaha jde spravnym smerem, ale je nutne se dobrat k porovnani CPI pred a po zmene a zrychleni by melo vyjit kolem 2%.

Uloha 12 [1]: 0.5-1b

- Bylo by dobre zduraznit rozdíl v rychlostech, zasadnim rozdilu ve zpusobu pristupu k datum (adresace, moznost selhani, apod.) a to, ze toto resi OS.

Uloha 13 [1]: 0.5-1b

- Proc se snizuje cetnost conflict missu?
- Tagy se porovnavaji paralelne, takže neni problem v tom najít shodu, ale v tom, dostat na vystup data ze spravne cache line.

Uloha 14 [2]: 1.5-2b

- Vas první nakres znazornuje primo mapovanou cache, az ten druhy je spravne.
- Struktura adresy neodrazi architekturu cache.
 - cacheline je 4B, tj. potrebujete 2 bity na offset (OK)

- 8 položek se rozdeli do 4 množin, tj. potřebujete 2 bity na index
 - zbyvajících 12 bitů potřebujete na tag
- Rezie by měla být vztazena ke kapacitě cache.

Uloha 15 [2]: 2b

- Chybi popis semantiky stavu, operací a sbernicových transakcí.

Znamka: 16.5-18.5b $\rightarrow$ 1