

Zadanie číslo 5, príklad 43

Princípy počítačového inžinierstva

Richard Németh- 120976

Zadanie:

Napíšte program, ktorý bude simulovať pohyb hráča na šachovnici podľa obrázka. Ľavé horné políčko šachovnice má súradnice (riadok, stĺpec) = (1,1) a pravé spodné políčko má súradnice (5,5). Na šachovnici sa nachádza určitý počet pokladov a pascí. Hráč môže začínať hru na niektorom z voľných políček.

V dátach bude pred spustením programu uložená postupnosť čísel (každé číslo na samostatnej adrese) reprezentujúca trasu, ktorú hráč počas hry vykoná, a to nasledovným spôsobom. Pohyb smerom:

- severovýchod = 1,
- juhovýchod = 2,
- juhozápad = 3,
- severozápad = 4.

Hodnota 0 reprezentuje ukončenie postupnosti. Hráč má povolené prechádzať aj cez okraje šachovnice – ak sa napríklad nachádza na súradniciach (3,5) a vykoná pohyb smerom na severovýchod, jeho nové súradnice budú (2,1). Zistite, či počas vykonávania zadanej postupnosti hráč

a) našiel poklad. V tomto prípade by mal program ihneď skončiť a do registra číslo 6 uložte hodnotu 1.

b) stúpil na pascu. V tomto prípade by mal program ihneď skončiť a do registra číslo 6 uložte hodnotu 2.

c) nenastalo a) ani b), teda program načítal ukončovací znak 0.

Aktuálnu (a teda aj štartovaciu) riadkovú a stĺpcovú súradnicu hráča uchovávalte na adresách 300 a 301.

	1	2	3	4	5
1		T			
2			P		
3				P	
4					T
5					

P – Poklad, T – Pasca

Riešenie:

V obrázkoch je vynechaných veľa príkazov NOP, aby sme sa mohli hlavne sústrediť na inštrukcie.

#### Prvá časť:

label	opcode
zac	LW \$25,00a0(\$0)
	LW \$26,00b0(\$0)
	LW \$22,0000(\$20)
	ADDI \$20,\$20,0004
	NOP
	BEQ \$22,\$1,pohyb1
	BEQ \$22,\$2,pohyb2
	BEQ \$22,\$3,pohyb3
	BEQ \$22,\$4,pohyb4
	BEQ \$22,\$0,koniec

Z dát si uložíme do registra začiatočné pozície hráča na 25,26. Následne do registra 22 načítame hodnotu kroku z dát posunutú o hodnotu uloženú v registri na pozícií 20. Pri začiatku každého cyklu je táto hodnota posunutá o 4 bity takže už načítavame nový krok. Následne sa len rozhodujeme o aký krok sa bude jednať.

#### Druhá časť:

label	opcode
pohyb1	ADDI \$25,\$25,0001
	SUBI \$26,\$26,0001
	BEQ \$0,\$0,kont1
	NOP
	NOP
	NOP
	NOP
	NOP
	NOP
	NOP
pohyb2	ADDI \$25,\$25,0001
	ADDI \$26,\$26,0001
	BEQ \$0,\$0,kont1
	NOP
	NOP
	NOP
	NOP
	NOP
	NOP
	NOP
pohyb3	SUBI \$25,\$25,0001
	ADDI \$26,\$26,0001
	BEQ \$0,\$0,kont1
pohyb4	SUBI \$25,\$25,0001
	SUBI \$26,\$26,0001
	BEQ \$0,\$0,kont1

V druhej časti iba meníme pozíciu hráča teda hodnotu registrov 25,26. Táto hodnota je pomocou odčítania alebo pričítania konštanty. Po ktoromkoľvek kroku sa presúvame do časti 3, kde sa vykonáva kontrola hráčovej pozície.

### Tretia časť:

kont1	BNEQ \$25,\$5,kont2
	SUBI \$25,\$25,0005
	BEQ \$0,\$0,kont2
	NOP
	NOP
	NOP
	NOP
	NOP
	NOP
	NOP
kont2	BNEQ \$26,\$5,kont3
	SUBI \$26,\$26,0005
	BEQ \$0,\$0,kont3
	NOP

label	opcode
kont3	NOP
	BNEQ \$26,\$0,kont4
	ADDI \$26,\$26,0005
	BEQ \$0,\$0,kont4
	NOP
	NOP
	NOP
	NOP
	NOP
	NOP
kont4	BNEQ \$25,\$0,poklad1
	ADDI \$25,\$25,0005
	BEQ \$0,\$0,poklad1

V tretej časti programu zisťujeme či hráč nie je mimo šachovnice. V registri na adrese 5 je uložená hodnota 6. Najprv skontrolujeme obe, alebo jedna z pozícií či nemá hodnotu 6 ak áno tak sa odpočítava číslo 5. Po každej kontrole skočí program na ďalšiu kontrolu takže skontroluje všetky prípady všetkých pozícií. Kontrola 1 a 2 sa pozerá na hodnotu 6, zvyšné dve kontroly porovnávajú hodnotu 0. V kontrole 4 sa presúvame na štvrtý krok, kde si overujeme poklad alebo pascu.

### Štvrtá časť:

label	opcode	label	opcode
poklad1	LW \$8,00c0(\$0)	poklad2	NOP
	LW \$9,00c0(\$4)		LW \$8,00d0(\$0)
	NOP		LI \$9,0004
	NOP		NOP
	BNEQ \$8,\$25,poklad2		BNEQ \$8,\$25,pasca1
	BNEQ \$9,\$26,poklad2		BNEQ \$9,\$26,pasca1
	BEQ \$0,\$0,poklad		BEQ \$0,\$0,poklad

V dátach sú uložené pozície pokladov a pascí tieto pozície sú prepísané do registra. Pozíciu x predstavuje register 8 a pozíciu y register 9. Následne by stačilo vynechať jeden NOP ale pre istotu som vynechal 2. Nasleduje porovnávanie pozícií pokiaľ sa pozícia nerovná program automaticky skáče na porovnávanie nového objektu. Výnimka je pri druhej pasci kde sa program vracia na začiatok cyklu. Pokiaľ sa obe hodnoty rovnajú, čo znamená že program preskočí obe inštrukcie BNEQ, tak sa presunie na svoj potrebný záver čiže do piatej časti.

Richard Németh  
120976

label	opcode	label	opcode
pasca1	LW \$8,00e0(\$0)	pasca2	LW \$8,00f0(\$0)
	LW \$9,00e0(\$4)		LW \$9,00f0(\$4)
	NOP		NOP
	NOP		NOP
	BNEQ \$8,\$25,pasca2		BNEQ \$8,\$25,zac
	BNEQ \$9,\$26,pasca2		BNEQ \$9,\$26,zac
	BEQ \$0,\$0,pasca		BEQ \$0,\$0,pasca
	NOP		NOP

Piata časť:

label	opcode
poklad	NOP
	NOP
	LI \$6,0001
	BEQ \$0,\$0,koniec
	NOP
	NOP
	NOP
	NOP
pasca	LI \$6,0002
	BEQ \$0,\$0,koniec
	NOP
	NOP
koniec	NOP
	NOP

Piata časť sa venuje iba ukončeniu programu. Pokiaľ hráč narazil na poklad uloží sa do registra 6 hodnota 1, ak by narazil na pascu uloží sa 2ka a presunie sa na koniec. Ak by dostal na začiatku krok s hodnotou 0 presunie sa na koniec.

## Opis Registra a Dát

The image shows two windows from a MIPS simulator: "Register" and "data memory".

**Register Window:** A table of 32 registers (R0 to R31). Handwritten annotations include:

- Registers R0-R5 are highlighted in green, labeled "Čísla pre Porovnávanie" (Numbers for comparison).
- Register R6 is highlighted in blue, labeled "Výsledok" (Result).
- Registers R8 and R9 are highlighted in orange, labeled "Pozícia [x] objektu" and "Pozícia [y] objektu" respectively.
- Registers R20 and R22 are highlighted in orange and red, labeled "OFFSET nového kroku" and "Momentálny krok" respectively.
- Registers R25 and R26 are highlighted in blue, labeled "Pozícia [x] hráča" and "Pozícia [y] hráča" respectively.

**Data Memory Window:** A table with columns for address and data. Handwritten annotations include:

- A red bracket groups rows 00000000 to 0000009f, labeled "Kroky hráča" (Player steps).
- Row 000000a0 is highlighted in orange, labeled "Pozícia [x] hráča" (Player's x position).
- Row 000000b0 is highlighted in orange, labeled "Pozícia [y] hráča" (Player's y position).
- A blue bracket groups rows 000000c0 to 000000ff, labeled "1 riadok = 1 objekt" (1 row = 1 object).
- Below the data memory, columns are labeled "[x]" and "[y]" with vertical lines, indicating the x and y coordinates of objects.

Pri dátach si môžeme modrou farbou všimnúť zapísané údaje. Tieto údaje predstavujú pozície objektu čiže pasce alebo pokladu. 1 riadok predstavuje 1 objekt, v prvom stĺpci majú uloženú pozíciu X a v druhom stĺpci pozíciu Y. Tieto hodnoty sú následne prepisované do registra 8,9 kde sa následne porovnávajú s momentálnou pozíciou hráča.

### Záver:

Program som sa snažil pretestovať rovnakou postupnosťou krokov ako v predošlom projekte. Problém bol že sa mi vždy program MipSim zasekol a prestal reagovať. Pokiaľ som testoval kratšiu postupnosť všetko mi fungovalo. Medzi inštrukciami sa nachádza zbytočne veľa inštrukcii NOP.

Richard Németh  
120976

Samotný program bol v mojom prípade problémový, pri prvom otvorení programu mi nechcel ukladať novo zapísané údaje, prepisoval mi inštrukcie, po zadání Label mi ho hneď zmazal či často krát zamrzol.