## Zadanie pre MIPSIM

## Princípy počítačového inžinierstva 2021

### Norbert Matuška

### 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 nachádzajú statické šachové figúrky, ktoré ohrozujú hráča, ak tento sa nachádza na určitých políčkach šachovnice (viď. obrázok). Ďalej sa tu môžu nachádzať prekážky, ktoré bránia figúrkam vo výhľade. Hráč môže začínať hru na niektorom z voľných políčok mimo ohrozenia figúrkou a môže vykonávať kroky o 1 políčko smerom na niektorú svetovú stranu. Hráč má povolené prechádzať aj cez okraje šachovnice – ak sa napríklad nachádza na súradniciach (2,5) a vykoná krok vpravo, jeho nové súradnice budú (2,1).

Od adresy 0h bude v pamäti údajov pred spustením programu uložená postupnosť čísel (každé číslo na samostatnej adrese) reprezentujúca trasu, ktorú hráč počas hry vykoná, a to nasledovným spôsobom. Pohyb smerom:

* hore = 1h,
* vpravo = 2h,
* dole = 3h,
* vľavo = 4h,
* koniec = 0h.

Hodnota 0h reprezentuje ukončenie postupnosti. Môžete predpokladať, že hráč sa nikdy nepokúsi vstúpiť na políčko s prekážkou, ani na políčko so šachovou figúrkou. Zistite, či počas vykonávania zadanej postupnosti dôjde alebo nedôjde k ohrozeniu hráča šachovou figúrkou. Ak áno, program by mal ihneď skončiť a na adresu c0h uložte hodnotu 1. Ak nie, teda program načíta ukončovací znak 0h, uložte na adresu c0h hodnotu 0. Aktuálnu (a teda aj štartovaciu) riadkovú a stĺpcovú súradnicu hráča uchovávajte na adresách a0h resp. b0h.

**Úloha 14**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 |
| 1 |  | • |  | • |  |
| 2 |  |  | S |  |  |
| 3 |  | • |  | • |  |
| 4 | • |  |  |  | • |
| 5 |  |  |  |  |  |

S – strelec, • - ohrozenie

### Riešenie

### Pamäť programu

|  |  |  |  |
| --- | --- | --- | --- |
| Adr. | Label | Inštrukcia | Komentár |
| 0h |  | LW $25,00a0($0) | Načítame začiatočnú riadkovú súradnicu z PÚ z adresy a0h do registra R25 |
| 4h |  | LW $26,00b0($0) | Načítame začiatočnú stĺpcovú súradnicu z PÚ z adresy b0h do registra R26 |
| 8h | begin | LW $22,0000($20) | Do registra R22 načítame prvok postupnosti z PÚ z adresy, na ktorú ukazuje ukazovateľ v registri R20 |
| ch |  | ADDI $20,$20,0004 | Zväčšíme ukazovateľ v registri R20 o 4, aby ukazoval na ďalší prvok postupnosti v poradí |
| 10h |  | NOP |  |
| 14h |  | BEQ $25,$1,cmp1 | Skontrolujeme, či hráč nestojí na riadku, kde sa nachádza ohrozenie, ak áno, skoč na kontrolu stĺpcovej súradnice cmp1 |
| 18h |  | BEQ $25,$3,cmp1 | Skontrolujeme, či hráč nestojí na riadku, kde sa nachádza ohrozenie, ak áno, skoč na kontrolu stĺpcovej súradnice cmp1 (2x môžeme využiť cmp1, pretože ohrozenia na riadku 1 a 3 sú v rovnakom stĺpci) |
| 1ch |  | BEQ $25,$4,cmp2 | Skontrolujeme, či hráč nestojí na riadku, kde sa nachádza ohrozenie, ak áno, skoč na kontrolu stĺpcovej súradnice cmp2 |
| 20h |  | BEQ $0,$0,start | Ak nie sme na riadkoch, kde sa nachádza ohrozenie, skáčeme na štart |
| 24h | cmp1 | BEQ $26,$2,gameover | Skontrolujeme, či hráč nestojí aj na stĺpci ohrozenia, ak áno, musíme ukončiť hru, teda skáčeme na gameover |
| 28h |  | BEQ $26,$4,gameover | Skontrolujeme, či hráč nestojí aj na stĺpci ohrozenia, ak áno, musíme ukončiť hru, teda skáčeme na gameover |
| 2ch |  | BEQ $0,$0,start | Ak nie sme na stĺpcoch, kde sa nachádza ohrozenie, skáčeme na štart |
| 30h | cmp2 | BEQ $26,$1,gamover | Skontrolujeme, či hráč nestojí aj na stĺpci ohrozenia, ak áno, musíme ukončiť hru, teda skáčeme na gameover |
| 34h |  | BEQ $26,$5,gameover | Skontrolujeme, či hráč nestojí aj na stĺpci ohrozenia, ak áno, musíme ukončiť hru, teda skáčeme na gameover |
| 38h |  | BEQ $0,$0,start | Ak nie sme na stĺpcoch, kde sa nachádza ohrozenie, skáčeme na štart |
| 3ch | gameover | SW $1,00c0($0) | Inkrementujeme údaj v data memory na adrese c0h o 1, ktorý nám podáva informáciu o tom, či sa hráč dostal do ohrozenia |
| 40h |  | SW $25,00a0($0) | Zapíšeme riadkovú súradnicu do data memory na adresu a0h |
| 44h |  | SW $26,00b0($0) | Zapíšeme stĺpcovú súradnicu do data memory na adresu b0h |
| 48h |  | BEQ $0,$0,end | A skočíme na koniec |
| 4ch |  | NOP |  |
| 50h | start | BEQ $22,$1,one | Ak je načítaný prvok postupnosti v reg. R22 rovný 1 (konštantu 1 máme uloženú v reg. R1) skoč na podprogram pre vykonanie pohybu hore ktorý sa nachádza na labeli „one“ |
| 54h |  | BEQ $22,$2,two | Ak je načítaný prvok postupnosti v reg. R22 rovný 2 (konštantu 2 máme uloženú v reg. R2) skoč na podprogram pre vykonanie pohybu vpravo ktorý sa nachádza na labeli „two“ |
| 58h |  | BEQ $22,$3,three | Ak je načítaný prvok postupnosti v reg. R22 rovný 3 (konštantu 3 máme uloženú v reg. R3) skoč na podprogram pre vykonanie pohybu dole ktorý sa nachádza na labeli „three“ |
| 5ch |  | BEQ $22,$4,four | Ak je načítaný prvok postupnosti v reg. R22 rovný 4 (konštantu 4 máme uloženú v reg. R4) skoč na podprogram pre vykonanie pohybu vľavo ktorý sa nachádza na labeli „four“  inak to musí byť 0 a pokračujeme ďalej: |
| 60h |  | SW $25,00a0($0) | Zapíšeme riadkovú súradnicu do data memory na adresu a0h |
| 64h |  | SW $26,00b0($0) | Zapíšeme stĺpcovú súradnicu do data memory na adresu b0h |
| 68h |  | BEQ $0,$0,end | A skočíme na koniec |
| 6ch | one | BEQ $25,$1,up5 | Je riadková súradnica rovná 1? ak hej, skáčeme na podprogram, kde sa posunieme na opačnú stranu hracej plochy |
| 70h |  | SUBI $25,$25,0001 | Ak nie, pohyb hore – znížime riadkovú súradnicu o 1 |
| 74h |  | BEQ $0,$0,begin | A skočíme naspäť na začiatok |
| 78h |  | NOP |  |
| 7ch | two | BEQ $26,$5,right5 | Je stĺpcová súradnica rovná 5? ak hej, skáčeme na podprogram, kde sa posunieme na opačnú stranu hracej plochy |
| 80h |  | ADDI $26,$26,0001 | Ak nie, pohyb vpravo – zvýšime stĺpcovú súradnicu o 1 |
| 84h |  | BEQ $0,$0,begin | A skočíme naspäť na začiatok |
| 88h |  | NOP |  |
| 8ch | three | BEQ $25,$5,down5 | Je riadková súradnica rovná 5? ak hej, skáčeme na podprogram, kde sa posunieme na opačnú stranu hracej plochy |
| 90h |  | ADDI $25,$25,0001 | Ak nie, pohyb dole – zvýšime riadkovú súradnicu o 1 |
| 94h |  | BEQ $0,$0,begin | A skočíme naspäť na začiatok |
| 98h |  | NOP |  |
| 9ch | four | BEQ $26,$1,left5 | Je stĺpcová súradnica rovná 1? ak hej, skáčeme na podprogram, kde sa posunieme na opačnú stranu hracej plochy |
| a0h |  | SUBI $26,$26,0001 | Ak nie, pohyb vľavo – znížime stĺpcovú súradnicu o 1 |
| a4h |  | BEQ $0,$0,begin | A skočíme naspäť na začiatok |
| a8h |  | NOP |  |
| ach | up5 | ADDI $25,$25,0004 | Zvýšime riadkovú súradnicu o 4, aby sme sa dostali na opačnú stranu hracej plochy |
| b0h |  | BEQ $0,$0,begin | A skočíme naspäť na začiatok |
| b4h |  | NOP |  |
| b8h | right5 | SUBI $26,$26,0004 | Znížime stĺpcovú súradnicu o 4, aby sme sa dostali na opačnú stranu hracej plochy |
| bch |  | BEQ $0,$0,begin | A skočíme naspäť na začiatok |
| c0h |  | NOP |  |
| c4h | down5 | SUBI $25,$25,0004 | Znížime riadkovú súradnicu o 4, aby sme sa dostali na opačnú stranu hracej plochy |
| c8h |  | BEQ $0,$0,begin | A skočíme naspäť na začiatok |
| cch |  | NOP |  |
| d0h | left5 | ADDI $26,$26,0004 | Zvýšime stĺpcovú súradnicu o 4, aby sme sa dostali na opačnú stranu hracej plochy |
| d4h |  | BEQ $0,$0,begin | A skočíme naspäť na začiatok |
| ... | ... | ... | ... |
| 140h | end | NOP | koniec programu |

Simulujeme napríklad takúto postupnosť krokov: 3, 2, 1, 4, 4, 4, 4. Začiatočné súradnice hráča sú (5,1). Konečné súradnice hráča po vykonaní postupnosti krokov budú (riadok, stĺpec) = (1,2) – podľa obrázka nižšie:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 |
| 1 |  | x• |  | • |  |
| 2 |  |  | S |  |  |
| 3 |  | • |  | • |  |
| 4 | • |  |  |  | • |
| 5 | H |  |  |  |  |

S – strelec, • - ohrozenie

### Obsah registrov a pamäti údajov pred spustením programu

Application, table

Description automatically generated

|  |  |  |
| --- | --- | --- |
| Register | Údaj | Komentár |
| R1 | 1h | konštanta 1 na porovnávanie |
| R2 | 2h | konštanta 2 na porovnávanie |
| R3 | 3h | konštanta 3 na porovnávanie |
| R4 | 4h | konštanta 4 na porovnávanie |
| R5 | 5h | konštanta 5 na porovnávanie |
| R20 | 0h | ukazovateľ do postupnosti prvkov, na začiatku ukazuje na 1. prvok |
| R22 | 0h | sem sa bude načítavať prvok postupnosti z pamäte údajov |
| R25 | 0h | sem sa načíta začiatočná riadková súradnica z PÚ z adresy a0h následne sa bude počas behu programu aktualizovať |
| R26 | 0h | sem sa načíta stĺpcová súradnica z PÚ z adresy b0h následne sa bude počas behu programu aktualizovať |

|  |  |  |
| --- | --- | --- |
| Adresa | Údaj | Komentár |
| 0h – 40h | 3h, 2h, 1h, 4h, 4h, …, 0h | postupnosť krokov |
| a0h | 3h | začiatočná riadková súradnica |
| b0h | 3h | začiatočná stĺpcová súradnica |
| c0h | 0h | začiatočná hodnota o ohrození hráča |

### Obsah registrov a pamäti údajov po spustení programu

Table

Description automatically generated

|  |  |  |
| --- | --- | --- |
| Register | Údaj | Komentár |
| R20 | 0ch | ukazovateľ do postupnosti prvkov, na konci ukazuje na prvok v postupnosti za poslednym, ktorý nás posunul do ohrozenia.(ukazuje preto už na ďalší, pretože máme tak program napísaný a skôr sa posunie ukazovateľ, ako sa vykoná kontrola pre ohrozenie hráča) |
| R25 | 1h | konečná riadková súradnica, ktorá sa uloží do PÚ na adresu a0h |
| R26 | 2h | konečná stĺpcová súradnica, ktorá sa uloží do PÚ na adresu b0h |

|  |  |  |
| --- | --- | --- |
| Adr. | Údaj | Komentár |
| 0h – 40h | 3h, 2h, 1h, 4h, 4h, …, 0h | postupnosť krokov |
| a0h | 1h | konečná riadková súradnica |
| b0h | 2h | konečná stĺpcová súradnica |
| c0h | 1h | konečná informácia o ohrození hráča |