

OPIS PROJEKTU

Maszyna wirtualna, to program emulujący procesor i pamięć. Przenośność języków takich jak Java czy Python jest wynikiem ich wykonywania na maszynie wirtualnej. Opiszemy najpierw prostą maszynę wirtualną.

Nasza maszyna wirtualna zawiera 64 (2^6) komórki pamięci (rejstry). W każdym rejestrze można zapisać int (4 bajtowa liczba ze znakiem). Oprócz tych rejestrów jest jeszcze rejestr flag oraz rejestr instrukcji. Na początku pracy rejestry są wyzerowane.

W opisywanej maszynie rejestr flag przechowuje informacje o tym, czy ostatnia operacja arytmetyczna (dodawanie, odejmowanie, mnożenie, dzielenie, porównywanie) zwróciła liczbę dodatnią (flaga D), ujemną (flaga U) albo zero (flaga Z). Ta informacja może być używana do instrukcji skoku.

Rejestr instrukcji informuje, którą z kolei instrukcję wykonujemy. Po wykonaniu instrukcji rejestr instrukcji zwiększa się o 1 (nie zawsze dotyczy instrukcji skoku).

Każda operacja, to liczba (unsigned) zapisana na dwóch bajtach (format little endian) i ewentualna liczba zapisana na czterech bajtach (format little endian).

Operacje (liczbę 16 bitową) można podzielić na 3 sekcje:

- najmłodsze 4 bity, to instrukcja jaką należy wykonać
- kolejne 6 bitów, to indeks pierwszego rejestru
- najstarsze 6 bitów, to indeks drugiego rejestru

R2 (6 bitów)	R1 (6 bitów)	Op (4 bity)
--------------	--------------	-------------

Dostępne instrukcje to:

Instrukcja	Opis	Op
dodaj	$R1 := R1 + R2$	0
odejmij	$R1 := R1 - R2$	1
mnóż	$R1 := R1 * R2$	2
dziel i oblicz resztę	$R1 := R1 / R2$ oraz $R2 := R1 \% R2$	3
porównaj	wynik $R1 - R2$ nie jest zapisywany jedynie zmiany w rejestrze flag	4
kopiuj	$R1 := R2$	5
instrukcja skoku	przeskok do innej instrukcji (opis poniżej, cała operacja 6 bajtowa)	6
wczytaj stałą	$R1 := 4$ bajty zapisane po tej operacji (cała operacja 6 bajtowa)	7
czytaj z klawiatury	do R1 wpisz liczbę z klawiatury	8
wpisz na ekran	z R1 wypisz liczbę na ekran	9
koniec	Zakończenie działania programu	10
nie używane		11-15

Za operacją skoku znajduje się liczba 4 bajtowa o jaką należy zmienić rejestr instrukcji. W R1 znajduje się informacja, kiedy należy wykonać operację skoku:

- 0 – zawsze
- 1 – gdy ustawiona jest flaga Z (zero)
- 2 – gdy nie jest ustawiona flaga Z (nie jest zero)
- 3 – gdy jest ustawiona flaga D (dodatnia)
- 4 – gdy jest ustawiona flaga U (ujemna)
- 5 – gdy są ustawione flagi D lub Z (nieujemna)
- 6 – gdy są ustawione flagi U lub Z (nieujemna)

Jeśli skok w operacji skoku nie jest wykonywany, to dodajemy 1 do rejestru instrukcji.

Napisz program w C++, który pomaga stworzyć i zapisać w pliku(binarnie) program na powyższą maszynę wirtualną (VM), oraz umożliwia uruchomienie zapisanego w pliku programu na VM. Program powinien umożliwiać uruchamianie programu na VM w trybie krokowym, czyli instrukcja po instrukcji, wypisując dodatkowe dane, takie jak wartości rejestrów.

Wpisz do programu kilka prostych przykładowych programów dla VM.

Przykład programu, który czyta z klawiatury liczby, wypisuje na ekran ich sumę i kończy pracę po wpisaniu 0:

- wczytaj liczbę z klawiatury do rejestru 0 (operacja $0 \cdot 2^4 + 8 = 8$)
- porównaj rejestr 0 z rejestrem 2 (operacja $2 \cdot 2^{10} + 0 \cdot 2^4 + 4 = 2052$), bo w rejestrze 2 jest wartość 0
- skocz o 2 instrukcje do przodu jeśli nie zero (operacja $2 \cdot 2^4 + 6 = 38$, i liczba 2)
- skończ program (operacja 10)
- do rejestru 1 dodaj rejestr 0, wynik do rejestru 1 (operacja $0 \cdot 2^{10} + 1 \cdot 2^4 + 0 = 16$)
- wypisz rejestr 1 (operacja $1 \cdot 2^4 + 9 = 25$)
- skocz 6 instrukcji do tyłu (operacja $0 \cdot 2^4 + 6 = 6$, i liczba -6)

Ten program jest zapisany na 22 bajtach (7 instrukcji po 2 bajty i 2 dodatkowe liczby po 4 bajty).