

8mu
Specyfikacja funkcjonalna

Krzysztof Piekarczyk
288277

25 marca 2020

Spis treści

1	Wstęp	2
1.1	Cel dokumentu	2
1.2	Cel projektu	2
1.3	Użytkownik końcowy	2
1.4	Uzasadnienie nazwy	2
2	Uruchomienie programu	3
3	Dane	4
3.1	Dane wejściowe	4
3.2	Dane wyjściowe	4
4	Sytuacje wyjątkowe	5
5	Testowanie	6

1 Wstęp

1.1 Cel dokumentu

Celem tego dokumentu jest przedstawienie funkcjonalności i obsługi programu "8mu,,.

1.2 Cel projektu

Celem projektu jest stworzenie interpretera języka CHIP-8, pozwalającego na uruchamianie napisanych w nim programów.

1.3 Użytkownik końcowy

Użytkownikiem końcowym są ludzie chcący zagrać w najstarsze gry, z których wiele zostało przeportowane na CHIP-8, ponadto każdy, kto chce spróbować swoich sił w pisaniu gier na CHIP-8.

1.4 Uzasadnienie nazwy

Jako, że interpreter CHIP-8 opiera swoją funkcjonalność na prostej maszynie wirtualnej, bardzo często jest używany jako pierwszy krok do świata emulacji. Postanowiłem więc nadać programowi nazwę "8mu,, która jest zlepkiem słów CHIP-8 i emulacja, czytane jako "ejt-mu,,.

2 Uruchomienie programu

Program uruchamiany będzie z linii komend: `./8mu <ścieżka>`. Na linii komend zostaną wypisane aktualne kroki w inicjalizacji programu, a po zakończonej inicjalizacji wywołane zostanie osobne okno, które służyć będzie za ekran interpretera CHIP-8.

Program może być uruchomiony również bez podania żadnego argumentu. Zostanie wtedy uruchomiony moduł diagnostyczny, który przeprowadzi testy funkcjonalności i poinformuje o ich wynikach.

3 Dane

3.1 Dane wejściowe

Danymi wejściowymi są pliki binarne zawierające kod zgodny ze specyfikacją języka CHIP-8. To rozwiązanie będzie implementowane w oparciu o te źródła:

- referencja techniczna: <https://github.com/mattmikolay/chip-8/wiki/CHIP-8-Technical-Reference>
- zestaw instrukcji: <https://github.com/mattmikolay/chip-8/wiki/CHIP-8-Instruction-Set>

Ponadto, użytkownik może wprowadzać bodźce dla interpretowanego programu poprzez klawisze "1, 2, 3, 4, Q, W, E, R, A, S, D, F, Z, X, C, V,, a więc siatkę 4x4 na klawiaturze.

3.2 Dane wyjściowe

Program wyświetla informację o inicjalizacji oraz ekran interpretera, który jest kontrolowany przez interpretowany program. Poza tym, program nie tworzy żadnych nowych plików.

4 Sytuacje wyjątkowe

Możliwe jest wystąpienie sytuacji, w której w pliku wejściowym, znajduje się kod operacyjny, który jest niepoprawny lub niezaimplementowany. W wypadku wyłapania takiego kodu na fazie inicjalizacji program zostanie przerwany i zostanie wyświetlona informacja o błędnym kodzie, wraz z pozycją tego kodu w pliku.

5 Testowanie

Zostanie zaimplementowany komponent diagnostyczny systemu który będzie sprawdzał, czy kody operacyjne są poprawnie wykonywane, będzie on jednak wymagał wizualnego potwierdzenia poprawnej funkcjonalności pod względem kodów dotyczących wyświetlania.