SPECYFIKACJA PROGRAMU

# Spis treści

1. [Informacje ogólne](#zakładka_0)
2. [Parametry wejściowe, sposób uruchamiania programu](#zakładka_1)
3. [opis przykładów](#zakładka_2)
4. [formatowanie pliku wejściowego](#zakładka_3)
5. [wyjście programu w przypadku prawidłowego zadziałania](#zakładka_4)
6. [wyjście programu w przypadku błędu](#zakładka_5)
7. [poruszanie się po programie](#zakładka_6)
8. [znaczenie kolorów w programie](#zakładka_7)

## Informacje ogólne

Zakładane są domyślne wymiary okna wiersza komend:

Wysokość nie mniejsza niż 33 linie

Szerokość nie mniejsza inż. 125 kolumn.

Program utworzony przy pomocy biblioteki graficznej nscurses

## Parametry wejściowe, sposób uruchamiania programu

Ścieżka względna(względem pliku .exe programu) jest **pierwszym** parametrem programu, zawsze występującym w jego wywołaniu.

W przypadku podania niepoprawnej ścieżki, lub jej braku program zwróci informację o konieczności jej ponownego wprowadzenia.

Jeżeli chcemy wprowadzić dodatkowe parametry, inne niż ścieżka pliku program musi zostać otwarty w konsoli.

**Drugim** argumentem programu może być psa\_code w przypadku wprowadzania kodu pseudoassemblera, bądź msck\_code w przypadku wprowadzania kodu maszynowego

**Trzecim** argumentem programu może być debug (tylko w przypadku gdy jest wprowadzony argument drugi) pozwala on przejść do graficznej formy egzekucji programu.

## opis przykładów

hexex.txt: Program realizuje to samo zadanie co dec\_to\_bin.txt tylko jest zapisany w kodzie maszynowym.

dec\_to\_bin.txt: Program wczytuje naturalną liczbę dziesiętną, a następnie zapisuje ją w tablicy w postaci naturalnego kodu binarnego w tablicy od końca.

factorial.txt: Program liczy silnię z 10 i zapisuje ją w pamięci.

errtest.txt: Celowo błędny przykład w którym występuje przepełnienie typu integer.

**Przykładowe wywołania programu:**

np: pseudoassembler-interpreter.exe ..\\przyklady\dec\_to\_bin.txt

np: pseudoassembler-interpreter.exe ..\\przyklady\errtest.txt psa\_code

np: pseudoassembler-interpreter.exe ..\\przyklady\dec\_to\_bin.txt msck\_code

np: pseudoassembler-interpreter.exe ..\\przyklady\hexex.txt msck\_code debug

np: pseudoassembler-interpreter.exe ..\\przyklady\dec\_to\_bin.txt psa\_code debug

## formatowanie pliku wejściowego

nie zakłada się komentarzy w pliku w kodzie maszynowym

sekcja danych oddzielona od sekcji rozkazów **dokładnie** jedną linią zawierającą dokładnie jeden znak "\n"

wolna linia pomiędzy sekcjami jest **jedyną** wolną linia jaką przewiduje się w pliku wejściowym

komentarze zajmujące całą linijkę **muszą** zaczynać się od znaków "//" na samym początku linii(kolumnie 0)

komentarze nie mogą być wprowadzane **bezpośrednio przed** sekcją rozkazów **ani bezpośrednio po** sekcji danych

wolna kolumna na początku linii oznacza brak etykiety w danym wierszu

## wyjście programu w przypadku prawidłowego zadziałania

Program zwraca w przypadku kodu maszynowego plik „var.txt” zawierający wyniki obliczeń(wypełnioną sekcję danych oraz końcowe stany rejestrów **ponieważ pisałem program przed ustaleniem nowej specyfikacji**)

Program zwraca w przypadku kodu pseudoassemblera 2 pliki „output.txt” oraz „var.txt” gdzie pierwszy zawiera podany kod w postaci maszynowej a drugi zawiera wyniki obliczeń te same co w przypadku kodu maszynowego.

Plik var.txt zawiera wartości losowe w miejscu niezadeklarowanej pamięci.

Program w przypadku skutecznego wykonania zwraca w konsoli „exec\_end”

## wyjście programu w przypadku błędu

Program przerywa działanie i zwraca błąd w przypadku wyjście poza sekcję danych w instrukcjach operujących na danych.

Program przerywa działanie i zwraca błąd w przypadku podzielenia przez zero.

Program przerywa działanie i zwraca błąd w przypadku przepełnienie zmiennej int.

Program przerywa działanie i zwraca błąd w przypadku wyjście poza sekcję rozkazów w instrukcjach typu jump.

Jeżeli program napotka błąd automatycznie zakończy wykonanie i zwróci komunikat o błędzie.

## poruszanie się po programie

Do przejścia do kolejnego rozkazu służy klawisz „x”

Do przerwania trybu debugowania i dokończenia obliczeń bez wyświetlania służy klawisz „q”

Do przewijania w górę dół(jeżeli któraś sekcja jest dłuższa niż 30 linii) służą:

„w” i „s” do przewijania sekcji rozkazów odpowiednio w górę i w dół

„e” i „d” do przewijania sekcji danych odpowiednio w górę i w dół

„r” i „f” do przewijania sekcji psudoassemlera odpowiednio w górę i w dół

## znaczenie kolorów w programie

Kolor żółty służy do zaznaczenia rozkazu który zostanie wykonany **jako następny** zarówno w sekcji rozkazów jak i w rejestrze stanu programu jak i w kodzie pseudoassemblera(jeżeli wystepuje).

Kolor czerwony służy do zaznaczenia komórki sekcji danych do której odwołuje się obecny rozkaz oraz drugiego półbajtu drugiego bajtu rozkazu gdy wskazuje on na sekcję danych.

Kolor zielony służy w instrukcji jump do zaznaczenia 1 lub 3 ostatnich bajtów instrukcji na którą wskazuje jump oraz sekcji danych samej instrukcji jump wraz z drugim półbajtem drugiego bajtu instrukcji jump.

Kolor niebieski służy w instrukcji rejestr-pamięć(innej niż jump) do zaznaczenia wykorzystywanego rejestru.

Kolor zielony i niebieski służą w instrukcji rejestr-rejest do zaznaczenia dwóch półbajtów drugiego bajtu i odpowiadających mu rejestrów. Jeżeli odwołanie jest do tego samego rejestru wykorzystywany jest tylko kolor zielony.

Kolor magenta i cyan służa do zaznaczenai obecnie wykonanej linii.