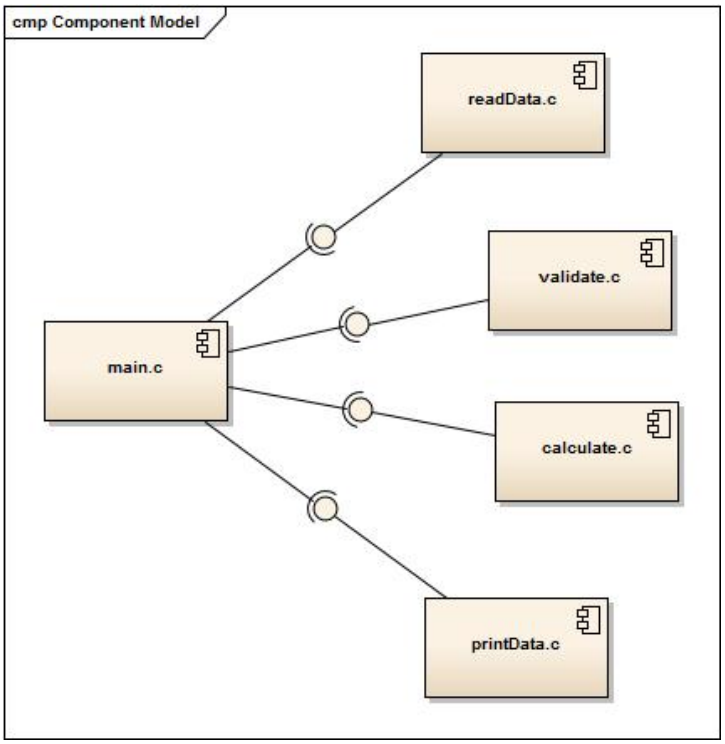
Specyfikacja Implementacyjna projektu Problem N-Ciał

Wojciech Świstowski (252572)

23 maja 2017

1 Podział programu na moduły

Program zostanie podzielony na 5 modułów, według poniższego schematu:



2 Funkcje programu

Implementacja projektu będzie zawierać 10 głównych funkcji, które dla przejrzystości kodu rozdzielone zostaną pomiędzy modułami według podobieństw funkcjonalności. Przyporządkowanie funkcji do modułów odbędzie się w poniższy sposób:

1. readData.c
    - Wczytywanie danych z plików wejściowych:

      ```
      struct body *readData(char **filename, int nFiles)
      ```

    - Zliczanie ciał:

      ```
      int checkBodyAmount(char **filename, int nFiles)
      ```

2. printData.c
    - Zapisywanie iteracji do plików wyjściowych:

      ```
      void printIterationToFile(char **fileNames, int size)
      ```

    - Wypisywanie wczytanych danych na ekran:

      ```
      void printData(struct body *dataBank, int size)
      ```

3. calculate.c
    - Obliczanie nowej pozycji ciała:

      ```
      void calculateNewPosition(int index, long timeDiff, int size)
      ```

    - Przeliczanie na sekundy czasu z parametrów wejściowych programu:

      ```
      long calculateTime(long value, char unit)
      ```

    - Obliczanie pozycji wszystkich ciał we wszystkich iteracjach:

      ```
      void calculateAllPositions(int size, long timeDiff)
      ```

4. validate.c
    - Sprawdzenie liczby plików wejściowych:

      ```
      void checkFileNumber(int fileNumber, int numberOfArguments)
      ```

    - Sprawdzenie czy wprowadzony argument wejściowy jest liczbą:

      ```
      void checkNumericArgument(char *argument)
      ```

    - Sprawdzenie kompletności danych:

      ```
      void checkForMissingData ()
      ```

3 Opis struktur

Do zaimplementowania programu użyta zostanie tylko jedna struktura danych:

```
struct body {
     char *name;
    double mass;
     double posX;
     double posY;
     double posZ;
     double velocityX;
     double velocityY;
     double velocityZ;
}
```

Struktura ta będzie przechowywała dane na temat ciał w obecnie obliczanej iteracji.

4 Dodatkowe biblioteki

Projekt będzie wykorzystywał poniższe biblioteki:

- stdio.h
- stdlib.h
- string.h
- math.h
- sys/stat.h

5 Testowanie

Podczas testowanie wykorzystywanych będzie przynajmniej kilka różnych plików wejściowych i wiele kombinacji argumentów wywołania programu. Sprawdzane będzie działanie programu dla: jednego, kilku plików wejściowych jak i braku takich plików (podane złe nazwy lub nie podano nazwy pliku wejściowego). Bardzo ważne będą testy dla przypadków, które mogą

wywołać nieprawidłowe działanie programu. Szczególną uwagę należy zwrócić na poniższe scenariusze:

- dane wejściowe dublujące się
- dane wejściowe niekompletne
- pusty plik wejściowy
- brak plików wejściowych
- nieprawidłowe argumenty wywołania.