

ĆWICZENIE 3 – PRZETWARZANIE PLIKÓW TEKSTOWYCH I BINARNYCH

Celem ćwiczenia jest nauczyć się tworzenia oraz odczytywania danych z plików 'csv' – 'Comma Separated Values' oraz tworzenia i odczytu plików binarnych. W trakcie ćwiczenia należy stworzyć klasy BinaryFile oraz CsvFile (nagłówki w plikach *.h, kod źródłowy w plikach *.cpp).

1. Obie klasy powinny mieć konstruktor z parametrami:
 - Ścieżka do pliku,
 - Tryb otwarcia pliku (zapisywanie, odczytywanie, nadpisanie),
2. Pliki powinny być otwierane w konstruktorze, i zamykane w destruktorze,
3. Obie klasy powinny zawierać metody:
 - read – metoda przyjmuje referencje na std::vector obiektów typu Point i zapisuje w nim punkty wczytane z pliku,
 - read – metoda przyjmuje referencję na obiekt typu Point oraz indeks punktu do wczytania i zapisuje w podanej referencji odczytany punkt z pliku o podanym indeksie, **bez odczytywania zawartości całego pliku.**
 - write – metoda przyjmuje referencje na std::vector obiektów typu Point i zapisuje jego zawartość do pliku,
4. Metody klas powinny zwracać kody błędów.
5. Przetestować działanie klas na plikach dostępnych na platformie.

Wskazówki:

1. Konieczne jest zapoznanie się z formatami plików binarnych oraz szablonami *Comma Separated Values*
2. Zapoznać się z działaniem klasy fstream:
<http://www.cplusplus.com/reference/fstream/fstream/>
3. Należy się zapoznać z przykładowym programem (*kod 3.1*)
4. Do wygenerowania chmury punktów można wykorzystać funkcje trójwymiarowe (np. $z(x, y) = \sin(x^2 + y^2)$) lub funkcję do generowania liczb losowych.
5. Zalogowane pliki csv można odczytać i wyświetlić na wykresach za pomocą programów Excel lub Matlab. W programie matlab można wyświetlić chmurę punktów dwoma instrukcjami:

```
points = csvread('points.csv');  
scatter3(points(:,1), points(:,2), points(:,3), 'r');
```

Uwagi:

- W metodach klas nie wolno używać funkcji typu : printf, scanf, std::cout, std::cin. Warstwa prezentacji musi zostać zaimplementowana w pętli głównej (poza klasą),
- W plikach nagłówkowych (*.h) nie wolno zawierać źródeł/rozwinień funkcji i metod klasowych (dotyczy to także konstruktorów i destruktorów),
- Powyższe uwagi obowiązują na wszystkich laboratoriach.