

Dokumentacja funkcjonalna

Jan Michorek, Anastasiia Prodius

March 23, 2025

Contents

1	Wstep	1
2	Technologie	1
3	Architektura systemu	1
4	Struktura kodu	2
5	Przykładowy kod	2
6	Podsumowanie	2

1 Wstep

Aplikacja do tworzenia grafów pozwala użytkownikowi wczytywać grafy z pliku o rozszerzeniu csrrg, a następnie wykonywać na nich pożądane operacje.

2 Technologie

Aplikacja została zbudowana przy użyciu następujących technologii:

- Język programowania: C
- Biblioteki: To be defined

3 Architektura systemu

Aplikacja składa się z modułów:

- Moduł generowania grafów
- Moduł wizualizacji
- Moduł analizy grafów
- Moduł wczytywania grafów z pliku

4 Struktura kodu

Kod źródłowy podzielony jest na następujące pliki i katalogi:

- src/
 - graph.c
 - csr_parser.c
- lib/
 - graph.h
 - csr_parser.h
- output/

5 Przykładowy kod

Do przechowywania grafów, służą tak zdefiniowane struktury:

Listing 1: struktura grafu

```
1 typedef enum {
2     UNDIRECTED,
3     DIRECTED
4 } GraphType;
5
6 typedef struct Node {
7     int id;
8     int ne;
9     struct Node **links;
10 } *Node;
11
12 typedef struct Graph {
13     GraphType type;
14     int n;
15     Node *nodes;
16 } Graph;
```

W bezpośredniej generacji grafów uczestniczą funkcje podane poniżej:

Listing 2: generacja

```
1 void link_nodes(Node, Node);
2 Node create_Node(int);
3 Graph * graph_init(int, GraphType type);
```

Zczytywanie z pliku o rozszerzeniu csrrg oraz zaczytywanie z macierzy.

Listing 3: parsowanie i konwertowanie

```
1 void parse_csrrg(FILE *in);
2 void parse_mat(FILE *in);
```

6 Podsumowanie

Aplikacja w podstawowej wersji zapewni użytkownikowi obsługę plików csrrg, i konwertowania sposobów zapisu grafów.