

Lista 5

Zadanie 1.

Zaprojektować i zaimplementować funkcje:

- obliczającą punkty przecięcia prostej danej równaniem kierunkowym $y = ax + b$, z okręgiem danym równaniem $(x - a)^2 + (y - b)^2 = r^2$, korzystając z definicji funkcji zadeklarowanych poniżej:
 - `int rozw_rown(SRownanie r, SPunkt& x1, SPunkt& x2)`
 - `int pkt_przec(SProsta& prosta, SOkrag& okrag, SPunkt& x1, SPunkt& x2)`
 - wykorzystującej wcześniej zdefiniowaną funkcję rozwiązującą równanie kwadratowe, (`rozw_rown`)
- orzekającą, czy dwie proste są do siebie prostopadłe (`bool czy_prostopadle(SProsta a, SProsta b)`).

W celu rozwiązania zadania utworzyć struktury opisujące:

- punkt,
- prostą,
- okrąg,
- równanie kwadratowe.

Zdefiniowane funkcje przetestować dla następujących danych wejściowych:

```
SProsta prosta1 = {1.0, 3.0};
SOkrag okrag = {0.0, 1.0, 2.0};
SProsta prosta2 = {-1.0, -33.2};
```

Zadanie 2.

Zaprojektować i zaimplementować strukturę opisującą *osobę*, z polami reprezentującymi jej *ID*, *e-mail*, *rok urodzenia* oraz *imię*. Zaprojektować i zaimplementować strukturę opisującą *listę kontaktów* o podanej nazwie (w pierwszej linijce pliku wejściowego znajduje się nazwa listy), pozwalającą przechować dowolną liczbę osób, umieszczonych w pliku z danymi wejściowymi (w drugiej linijce pliku wejściowego znajduje się liczba, która informuje o tym ile osób znajduje się na liście poniżej). Dla powyższych struktur zdefiniować funkcje:

- `wczytaj` – inicjalizującą listę kontaktów danymi wczytanymi z pliku,
- `wypisz` – pozwalającą wysłać na zadany (standardowy/plikowy) strumień wszystkie osoby z podanej listy kontaktów, z zachowaniem formatu pliku wejściowego,
- `wypisz` – pozwalającą wysłać na zadany (standardowy/plikowy) strumień dane pojedynczej osoby, z zachowaniem formatu pliku wejściowego,
- `szukaj` – realizującą wyszukanie na liście kontaktów osoby o podanym (z linii wywołania programu) numerze gadu-gadu oraz zwracającą dane tej osoby, które można odczytać na standardowym wyjściu z wykorzystaniem funkcji `wypisz`

- **podzial** – realizującą zapisanie danych osobowych na dwa różne strumienie plikowe (przekazane do funkcji jako parametry), z podziałem ze względu na płeć (na podstawie ostatniej litery imienia), nazwy plików pobrać z linii wywołania programu, do zapisu danych należy wykorzystać funkcję **wypisz**
- **usun** – zwalnijącą zasoby przydzielone na stercie.

Zadanie 3.

Zaprojektować i zaimplementować strukturę opisującą *studenta*, z polami reprezentującymi jego *imie*, *nazwisko* oraz *liczbę i wykaz niezaliczonych kursów*. Zaprojektować i zaimplementować strukturę opisującą *listę studentów* pozwalającą przechować dowolną liczbę studentów, umieszczonych w pliku z danymi wejściowymi. Dla powyższej struktury zdefiniować funkcje:

- **wczytaj** – inicjalizującą listę studentów danymi wczytanymi z pliku i zwracającą wykaz wszystkich studentów utworzony wewnątrz tej funkcji,
- **wypisz** – pozwalającą wyświetlić na standardowym wyjściu pełne informacje o studentach znajdujących się na liście,
- **usun** – zwalnijącą zasoby przydzielone na stercie.