

# Część I. Aplikacja konsolowa

Napisz program implementujący algorytm przeszukiwania tablicy z wartownikiem. Opis algorytmu znajduje się poniżej.

By odnaleźć element  $x$  podejmiemy następujące kroki:

1. na końcu tablicy (pod indeksem  $n+1$ ) wstawimy szukany element  $x$  - będzie to nasz wartownik, w przypadku, gdy nie znajdziemy go nigdzie indziej w tablicy, zabezpieczy nas on przed wyjściem poza tablicę,
2. przejdziemy po kolejnych elementach tablicy, tak długo aż nie znajdziemy szukanego elementu,
3. w momencie znalezienia szukanego elementu  $x$  sprawdzamy, który jest to element tablicy? Jeżeli jest to ostatni element tablicy ( $n+1$ ) to trafiliśmy na naszego wartownika i oznacza to, że w tablicy nie było szukanego elementu  $x$ , w przeciwnym razie element  $x$  został odnaleziony.

Założenia do programu:

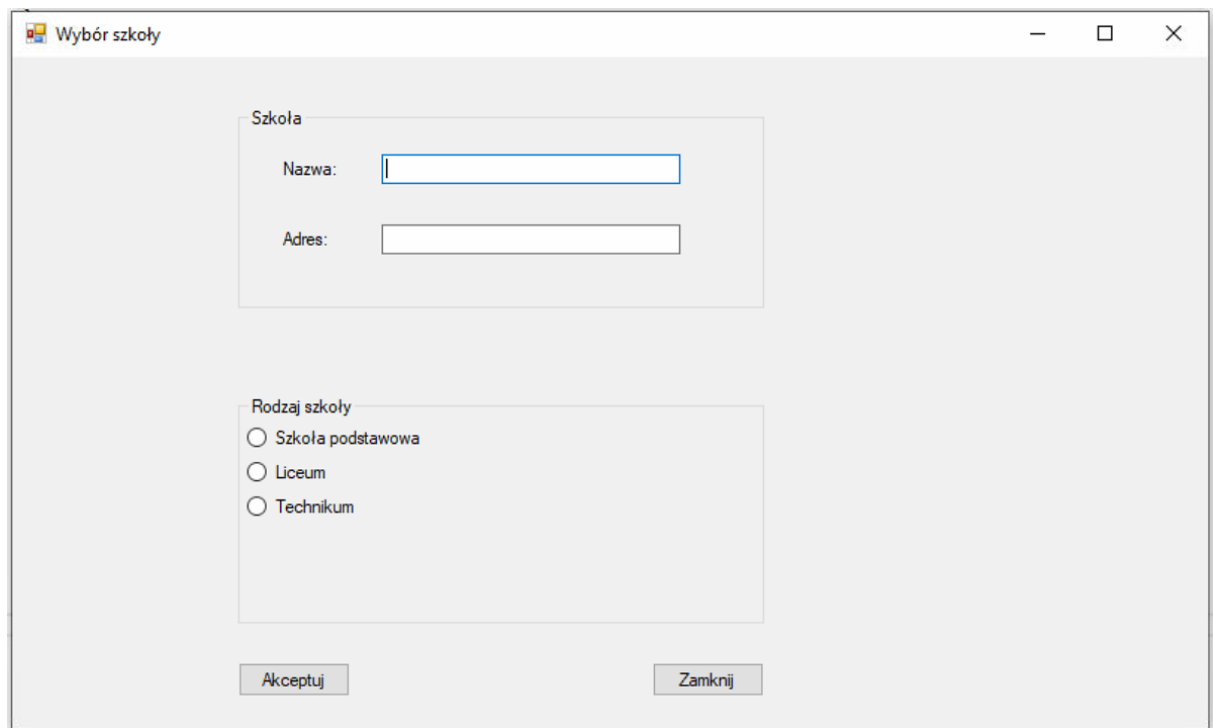
- Program wykonywany w konsoli.
- Zastosowany obiektowy język programowania zgodny z zainstalowanym na stanowisku egzaminacyjnym: C++ lub C#, lub Java, lub Python.
- Przeszukiwana jest minimum 50 elementowa tablica liczb całkowitych wypełniona wartościami pseudolosowymi z zakresu od 1 do 100.
- Wyszukiwane jest pierwsze wystąpienie elementu w tablicy.
- Wypełnianie tablicy i przeszukiwanie jest realizowane w oddzielnych funkcjach / metodach. Funkcja przeszukująca zwraca indeks odnalezionego elementu.
- Wartość do wyszukania jest pobierana z klawiatury.
- Po przeszukaniu tablicy na ekranie wyświetlana jest zawartość tablicy (liczby oddzielone przecinkami) oraz indeks, pod którym odszukano wartość lub w przypadku jej braku – stosowny komunikat.
- Program powinien podejmować zrozumiałą komunikację z użytkownikiem, dane wprowadzane i wyprowadzane powinny być opatrzone zrozumiałym opisem.
- Program powinien być zapisany czytelnie, z zachowaniem zasad czystego formatowania kodu, należy stosować znaczące nazwy zmiennych i funkcji.

Podejmij próbę kompilacji i uruchomienia. Informacje dotyczące dokumentacji i zrzutów ekranowych umieszczono w części IV zadania.

Kod aplikacji przygotuj do nagrania na płytę. W podfolderze *konsolowa* powinno znaleźć się archiwum całego projektu o nazwie *konsolowa.zip*

## Część II. Aplikacja okienkowa

Napisać program, który odtworzy następujący wygląd okna:



### Opis wyglądu aplikacji. Elementy od góry:

Okno zawiera dwie ramki grupujące (*Group Box*).

Pierwsza ramka podpisana u góry jako „Szkoła”, druga jako „Rodzaj szkoły”

Pierwsza ramka zawiera dwie kontrolki Label z napisami „Nazwa” oraz „Adres”.  
Pierwsza ramka zawiera również dwa pola tekstowe TextBox.

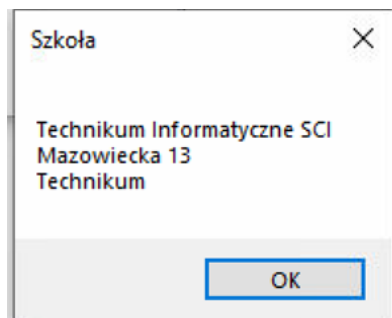
Druga ramka zawiera trzy pola wyboru (*RadioButton*). Z opisami jak na powyższym rysunku.

Poniżej znajdują się dwa przyciski: Akceptuj oraz Zamknij, umieszczone na tym samym poziomie.

Wszystkie elementy widoczne w oknie aplikacji muszą być rozmieszczone w pionie oraz poziomie jak na powyższym rysunku czyli np. Label „Nazwa” oraz znajdujące się obok pole TextBox umieszczone są na tym samym poziomie.

### Działanie aplikacji

Po wybraniu przez użytkownika przycisku **Akceptuj**, nazwa, adres oraz rodzaj szkoły powinny zostać zaprezentowane w oknie informacyjnym MessageBox jak na poniższym rysunku.



Okno MessageBox powinno mieć tytuł „Szkoła”

Po wybraniu przycisku Zamknij, program powinien zakończyć swoje działanie.

Założenia do programu

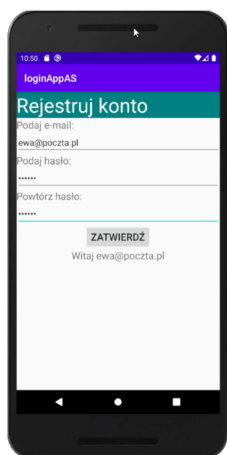
- Program wykonywany w oknie.
- Obiektowy język programowania zgodny z zainstalowanym na stanowisku egzaminacyjnym: C++ lub C# lub Java lub Python.
- Program powinien być zapisany czytelnie, z zasadami czystego formatowania kodu, należy stosować znaczące nazwy zmiennych, funkcji i kontrolek.

Podejmij próbę kompilacji i uruchomienia. Informacje dotyczące dokumentacji i zrzutów ekranowych umieszczono w części IV zadania.

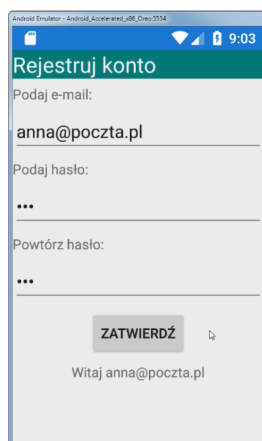
Kod aplikacji przygotuj do nagrania na płytę. W podfolderze *okenkowa* powinno znaleźć się archiwum całego projektu o nazwie *okienkowa.zip*

## Część III. Aplikacja mobilna

Wykonaj aplikację mobilną za pomocą środowiska programistycznego dostępnego na stanowisku egzaminacyjnym oraz uruchom ją w dostępnym emulatorze systemu mobilnego.



Obraz 1a.  
Emulacja na urządzeniu Nexus 5X API 29 x86.  
Aplikację utworzono w Android Studio



Obraz 1b.  
Emulacja na systemie Android Oreo. Aplikację utworzono w MS Visual Studio

Na obrazach 1a i 1b przedstawiono działanie aplikacji mobilnej. W zależności od użytego środowiska programistycznego oraz emulowanego systemu wynik końcowy może nieznacznie się różnić.

### Opis wyglądu aplikacji. Elementy od góry:

- Napis „Rejestruj konto”.
- Napis „Podaj e-mail:”, a pod nim pole edycyjne z podpowiedzią o treści „email”.
- Napis „Podaj hasło:”, a pod nim pole edycyjne do wprowadzenia hasła, realizuje ukrywanie hasła jak na obrazach 1a i 1b.
- Napis „Powtórz hasło:”, a pod nim pole edycyjne do wprowadzenia hasła, realizuje ukrywanie hasła jak na obrazach 1a i 1b.
- Przycisk o treści „ZATWIERDŹ”, jest on wyśrodkowany.
- Obszar do wyświetlania komunikatów, jest on wyśrodkowany.

### Działanie aplikacji

- Po wybraniu przycisku ZATWIERDŹ jest sprawdzane:
  - Czy e-mail zawiera znak @.
  - Czy podane hasło jest równe powtórzonemu hasłu.
- W obszarze do wyświetlania komunikatów pojawia się napis:
  - Na początku działania aplikacji: „Autor”, dalej wstawiony numer PESEL zdającego.
  - Po zatwierdzeniu, gdy e-mail jest niepoprawny: „Nieprawidłowy adres e-mail”.
  - Po zatwierdzeniu, gdy hasła się różnią: „Hasła się różnią”.
  - Po zatwierdzeniu, gdy nie wystąpiły błędy: „Witaj <e-mail>”, gdzie <e-mail> oznacza aktualnie wprowadzony adres e-mail.

### Założenia aplikacji

- Interfejs użytkownika zapisany za pomocą języka znaczników wspieranego w danym środowisku (np. XAML, XML).
- Zastosowany typ rozkładu liniowy (Linear / Stack lub inny o tej idei).
- Zastosowany kolor tła dla napisu „Rejestruj konto”: Teal (#008080), zgodnie z Obrazem 1a.
- Zastosowany kolory czcionki: czarny i biały, zgodnie z Obrazem 1a.
- Czcionka napisu „Rejestruj konto” jest wizualnie większa od pozostałych.
- Pola edycyjne są rozciągnięte na całą szerokość ekranu.
- Aplikacja powinna być zapisana czytelnie, z zasadami czystego formatowania kodu, stosować znaczące nazwy zmiennych i funkcji.

Podjmij próbę kompilacji i emulacji. Informacje dotyczące dokumentacji i zrzutów ekranowych umieszczono w części IV zadania.

Kod aplikacji przygotuj do nagrania na płytę. W podfolderze *mobilna* powinno znaleźć się archiwum całego projektu o nazwie *mobilna.zip*

## Część IV. Dokumentacja aplikacji

Wykonaj dokumentację do aplikacji utworzonych na egzaminie. W kodzie źródłowym aplikacji konsolowej utwórz nagłówek funkcji przeszukującej, według wzoru zgodnie z listingiem 1. Nagłówek powinien znaleźć się w kodzie źródłowym nad funkcją. W miejscu znaków <> należy podać nazwę funkcji, nazwy argumentów wraz z krótkim opisem, typ zwracany wraz z krótkim opisem, krótki opis zawierający przynajmniej nazwę algorytmu. Gdy funkcja nie ma argumentów – zapisać „brak”. W miejscu autor należy podać swoje nazwisko i imię.

*UWAGA: Dokumentację umieścić w komentarzu (wieloliniowym lub kilku jednoliniowych). Znajdujący się w listingu 1 wzór dokumentacji jest bez znaków początku i końca komentarza, gdyż te są różne dla różnych języków programowania*

Listing 1. Wzór dokumentacji funkcji

```
*****
nazwa funkcji: <tu wstaw nazwę funkcji>
argumenty: <nazwa argumentu> - <co przechowuje>
           < nazwa argumentu> - <co przechowuje>
typ zwracany: <nazwa typu>, <co jest zwracane>
informacje: <opis>
autor: <Nazwisko i imię zdającego>
*****
```

Wykonaj zrzuty ekranu dokumentujące uruchomienie aplikacji utworzonych podczas egzaminu. Zrzuty powinny obejmować cały obszar ekranu monitora z widocznym paskiem zadań. Jeżeli aplikacja uruchamia się, na rzucie należy umieścić okno z wynikiem działania programu oraz otwarte środowisko programistyczne z projektem lub okno terminala z kompilacją projektu. Jeżeli aplikacja nie uruchamia się z powodu błędów kompilacji, należy na rzucie umieścić okno ze spisem błędów i widocznym otwartym środowiskiem programistycznym. Wykonać należy tyle zrzutów ile interakcji podejmuje aplikacja. Wymagane zrzuty ekranu:

- Aplikacja konsolowa – dowolna liczba zrzutów nazwanych *konsola1*, *konsola2* ...
- Aplikacja okienkowa – dowolna liczba zrzutów nazwanych *window1*, *window2* ...
- Aplikacja mobilna – dowolna liczba zrzutów nazwanych *mobile1*, *mobile2* ...

W edytorze tekstu pakietu biurowego utwórz plik z dokumentacją i nazwij go *egzamin*. Dokument powinien zawierać podpisane zrzuty ekranu oraz zapisane informacje:

- o Nazwę systemu operacyjnego, na którym pracował zdający.
- o Nazwy środowisk programistycznych, z których zdający korzystał na egzaminie.
- o Nazwy języków programowania.
- o Nazwa emulatora systemu mobilnego.

Zrzuty ekranu i dokument umieścić w podfolderze *dokumentacja*.

*Nagraj folder z rezultatami pracy. W folderze z Nazwiskiem i imieniem zdającego powinny się znajdować podfoldery: konsola, okienkowa, mobilna, dokumentacja.*

*W folderze dokumentacja: pliki ze zrzutami oraz plik egzamin.*

*W folderze konsola: spakowany cały projekt aplikacji konsolowej, pliki z kodem źródłowym, opcjonalnie plik uruchomieniowy.*

*W folderze okienkowa: spakowany cały projekt aplikacji okienkowej, pliki z kodem źródłowym, opcjonalnie plik uruchomieniowy*

*W folderze mobilna: spakowany cały projekt aplikacji mobilnej, pliki ze źródłami interfejsu i logiki.*

Upewnij się, że wszystko co stworzyłeś zostało prawidłowo nagrane