

## Zadanie egzaminacyjne

**UWAGA:** katalog z rezultatami pracy oraz płytę należy opisać numerem zdającego, którym został podpisany arkusz, czyli numerem PESEL lub w przypadku jego braku numerem paszportu. Dalej w zadaniu numer ten jest nazwany numerem zdającego.

Wykonaj aplikację konsolową oraz desktopową według wskazań. Wykonaj dokumentację do aplikacji konsolowej, zgodnie z opisem w części III instrukcji do zadania. Do pracy w systemie operacyjnym wykorzystaj konto **Egzamin** bez hasła.

Utwórz folder i nazwij go numerem zdającego. W folderze utwórz podfoldery: *konsola*, *desktopowa*, *dokumentacja*. Po wykonaniu każdej aplikacji, jej pełny kod (cały folder projektu) **spakuj do archiwum**. Następnie pozostaw w podfolderze jedynie pliki źródłowe, których treść była modyfikowana, plik uruchomieniowy, jeśli jest to możliwe oraz spakowane archiwum.

### Część I. Aplikacja konsolowa

Za pomocą narzędzi do tworzenia aplikacji konsolowych zaimplementuj program sprawdzający poprawność numeru PESEL. Program powinien sprawdzać płeć i sumę kontrolną według opisu:

Numer PESEL jest to 11-cyfrowy identyfikator numeryczny.

#### Płeć

Informacja o płci osoby zawarta jest na 10. (przedostatniej) pozycji numeru PESEL.

- cyfry 0, 2, 4, 6, 8 (parzyste) – oznaczają płeć żeńską
- cyfry 1, 3, 5, 7, 9 (nieparzyste) – oznaczają płeć męską

#### Cyfra kontrolna i sprawdzanie poprawności numeru

Jedenastą cyfrą numeru PESEL jest cyfra kontrolna umożliwiająca kontrolę poprawności identyfikatora. Jest ona wynikiem działania na pierwszych dziesięciu cyfrach.

Algorytm obliczania cyfry kontrolnej na podstawie kolejnych cyfr numeru:

1. Dla kolejnych 10 cyfr numeru PESEL oblicz iloczyn każdej cyfry i jej wagi na podstawie tabeli:

Pozycja cyfry od lewej	1	2	3	4	5	6	7	8	9	10
Waga cyfry	1	3	7	9	1	3	7	9	1	3

Oznacza to, że pierwszą cyfrę numeru PESEL należy pomnożyć przez 1, drugą cyfrę przez 3, trzecią przez 7 itd.

2. Wszystkie iloczyny zsumuj ze sobą i zapisz w zmiennej S
3. Wykonaj operację modulo 10 na sumie S i zapisz w zmiennej M
4. Gdy wartość zmiennej M jest równa 0, to zmiennej R przypisz wartość 0. W przeciwnym przypadku zmiennej R przypisz wartość różnicy 10 i M ( $R=10-M$ )
5. Zmienna R stanowi sumę kontrolną numeru PESEL i musi być równa jedenastej cyfrze numeru PESEL

#### Założenia aplikacji:

- Zastosowany obiektowy język programowania zgodny z zainstalowanym na stanowisku egzaminacyjnym: C++ lub C#, lub Java, lub Python
- Numer PESEL może być przechowywany jako zmienna tekstowa albo tablica 11 liczb całkowitych lub znaków
- Zmienna ta jest zainicjowana numerem PESEL zdającego lub w przypadku jego braku numerem 55030101193
- Sprawdzanie płci należy zaimplementować w osobnej funkcji zwracającej typ znakowy o wartości 'K' dla kobiety oraz 'M' dla mężczyzny
- Sprawdzanie sumy kontrolnej należy zaimplementować w osobnej funkcji zwracającej wartość logiczną `true` w przypadku zgodności sumy lub `false` w przeciwnym przypadku
- Parametrem wejściowym obu funkcji jest zmienna przechowująca numer PESEL

- Program główny testuje działanie funkcji i zawiera następujące operacje wejścia – wyjścia
  - Wczytanie z klawiatury numeru PESEL
  - Wypisanie płci (ustalonej przez funkcję) w postaci napisu: „Kobieta” lub „Mężczyzna” w oparciu o wczytany numer PESEL
  - Wypisanie informacji o zgodności lub niezgodności sumy kontrolnej w oparciu o wczytany numer PESEL
- Program powinien podejmować zrozumiałą komunikację z użytkownikiem
- W programie może być zastosowane angielskie lub polskie nazewnictwo zmiennych i funkcji
- Program powinien być zapisany czytelnie, z zachowaniem zasad czystego formatowania kodu, należy stosować nazwy zmiennych znaczące oraz zgodne z przedstawionym algorytmem
- Do kodu należy dołączyć dokumentację, która została opisana w części III zadania egzaminacyjnego.

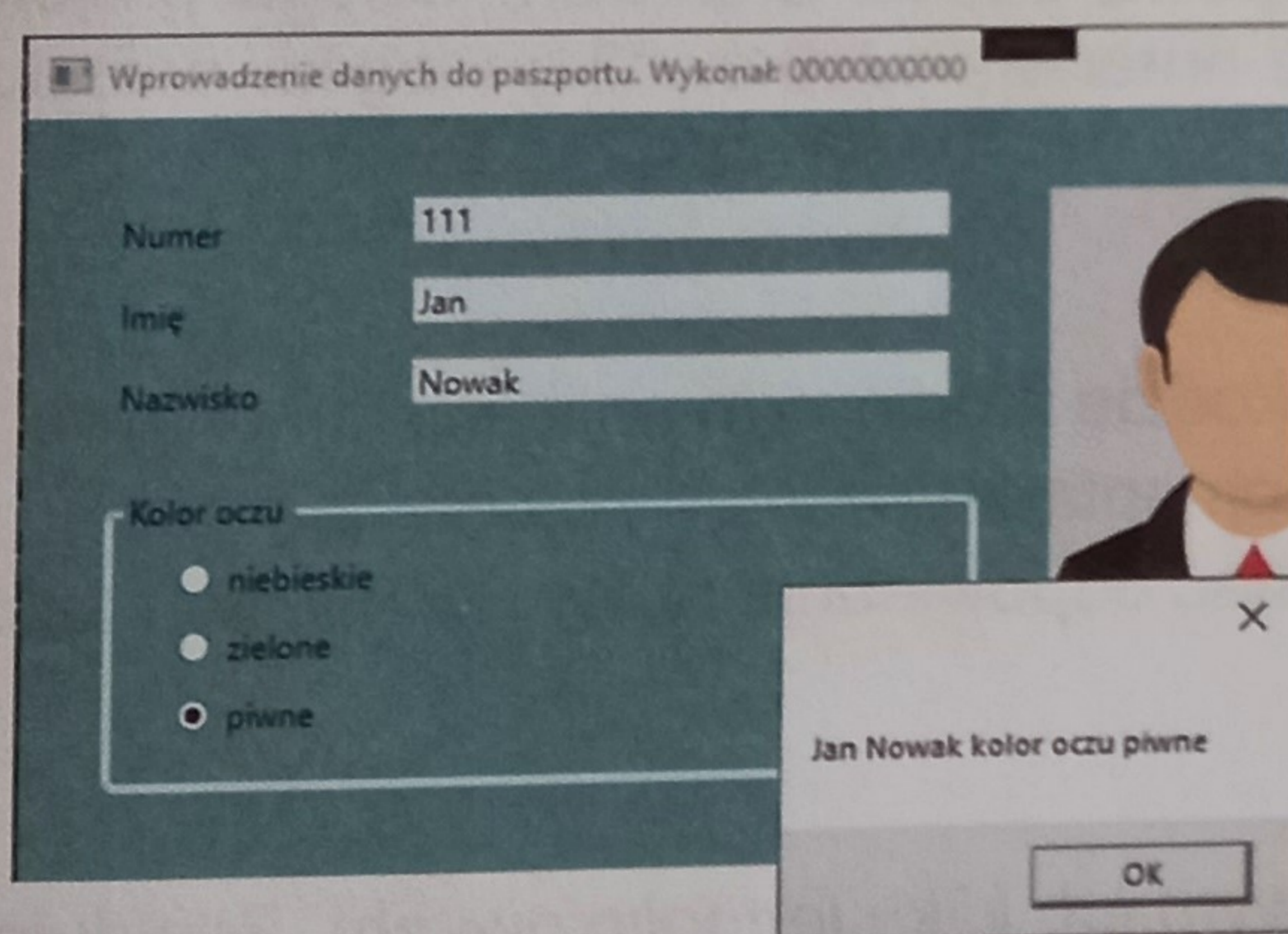
Kod aplikacji przygotuj do nagrania na płytę. W folderze *konsola* zapisz archiwum całego projektu o nazwie *konsola.zip*, plik z kodem źródłowym programu oraz plik uruchomieniowy, jeżeli istnieje.

## Część II. Aplikacja desktopowa

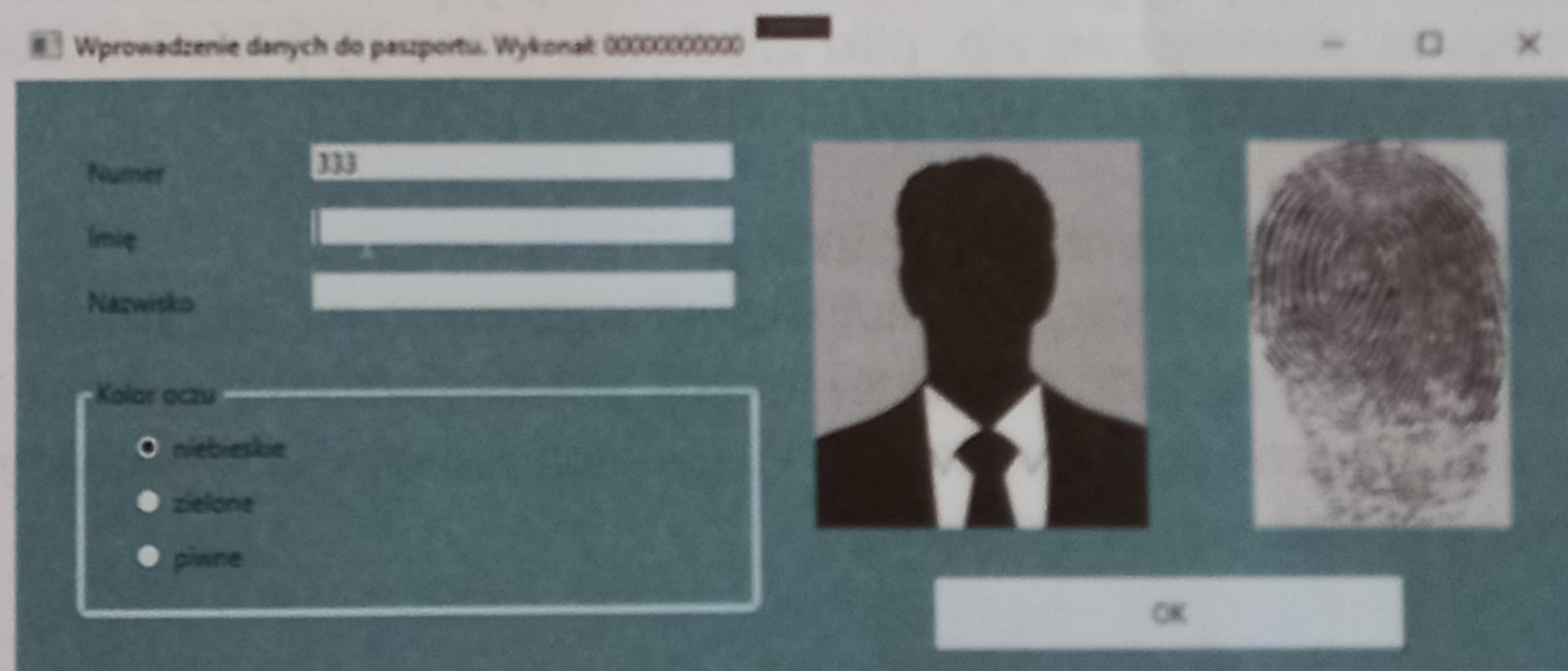
Za pomocą dostępnego na stanowisku egzaminacyjnym środowiska programistycznego wykonaj aplikację desktopową do wprowadzania danych paszportowych. Do wykonania zadania wykorzystaj znajdujące się na pulpicie archiwum 7-Zip o nazwie *materialy.7z* zabezpieczone hasłem: **p@szporTy%**



Obraz 1. Stan początkowy aplikacji



Obraz 2. Fragment okna po wybraniu przycisku „OK”



Obraz 3. Po opuszczeniu pola edycyjnego „Numer”

Na obrazie 1 przedstawiono ideę aplikacji desktopowej. W zależności od użytego środowiska programistycznego wygląd może nieznacznie się różnić.

### Opis wyglądu aplikacji

- Okno o nazwie „Wprowadzenie danych do paszportu. Wykonak: ” następnie wstawiony numer zdającego
- Kontrolki rozmieszczone zgodnie z obrazem 1:
  - Pole edycyjne poprzedzone etykietą o treści „Numer”
  - Pole edycyjne poprzedzone etykietą o treści „Imię”
  - Pole edycyjne poprzedzone etykietą o treści „Nazwisko”

- Grupa „Kolor oczu” zawierająca trzy pola wyboru: „niebieskie”, „zielone”, „piwne”. Pierwsze pole jest domyślnie zaznaczone
- W grupie może być jednocześnie zaznaczone jedno pole
- Przycisk o treści „OK”
- Dwa obrazy: 000-zdjecie.jpg oraz 000-odcisk.jpg. Obrazy mają tę samą wysokość. W aplikacji, której okno przedstawiono na obrazie 1 zastosowano wysokość równą 180
- Okno ma tło koloru CadetBlue (#5F9EA0)
- Pola edycyjne i przycisk mają tło koloru Azure (#F0FFFF)

Działanie aplikacji po opuszczeniu pola edycyjnego „Numer”

- Aktualizowane są oba zdjęcia w oknie. Nazwy plików graficznych są utworzone na podstawie wpisanego numeru do pola edycyjnego „Numer”
- Obraz osoby ma nazwę <numer>-zdjecie.jpg, gdzie <numer> został pobrany z pola edycyjnego, np. po wpisaniu do pola edycyjnego „333” ustawiona nazwa zdjęcia to 333-zdjecie.jpg
- Podobnie, obraz odcisku palca ma nazwę <numer>-odcisk.jpg, gdzie <numer> został pobrany z pola edycyjnego, np. 333-odcisk.jpg (Obraz 3)
- Do testów aplikacji należy wykorzystać wszystkie obrazy z wypakowanego archiwum. W przypadku wpisania numeru (np. 444), który nie odpowiada żadnemu plikowi graficznemu, obraz nie jest wyświetlany.

Działanie aplikacji po wybraniu przycisku „OK”

- Jeżeli dane zostały wprowadzone do wszystkich pól edycyjnych, wyświetlany jest komunikat zgodny z obrazem 2, o treści: „<imie> <nazwisko> kolor oczu <kolor>”, gdzie pola w nawiasach <> zostały pobrane z kontrolek
- Jeżeli nie wpisano imienia lub nazwiska, wyświetlany jest komunikat „Wprowadź dane”

Aplikacja powinna być zapisana czytelnie, z zasadami czystego formatowania kodu, należy stosować znaczące nazwy zmiennych i funkcji.

Podjmij próbę kompilacji i uruchomienia aplikacji. Informacje dotyczące dokumentacji i zrzutu ekranowego umieszczono w części III zadania egzaminacyjnego.

Kod aplikacji przygotuj do nagrania na płytę. W folderze *desktopowa* zapisz archiwum całego projektu o nazwie *desktopowa.zip*, pliki źródłowe, które były modyfikowane (interfejs użytkownika i logika aplikacji) oraz plik uruchomieniowy, jeżeli istnieje.

### Część III. Dokumentacja aplikacji

Wykonaj dokumentację do aplikacji utworzonych na egzaminie. W kodzie źródłowym aplikacji konsolowej utwórz komentarz do dowolnej funkcji, według wzoru z listingu 1. Komentarz powinien znaleźć się nad funkcją lub pod jej nagłówkiem. W miejscu nawiasów <> należy podać odpowiednie opisy. W miejscu autor należy podać numer zdającego.

**UWAGA:** Dokumentację należy 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 (liczba gwiazdek dowolna)

```
*****
nazwa funkcji:      <nazwa>
opis funkcji:      <krótki opis co robi funkcja>
parametry:         <nazwa parametru - opis >
zwracany typ i opis: <nazwa typu i opis co jest zwracane>
autor:             <numer 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 zrzucie 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 zrzucie umieścić okno ze spisem błędów i widocznym otwartym środowiskiem programistycznym. Należy wykonać tyle zrzutów, ile interakcji podejmuje aplikacja (np. stan początkowy, po wpisaniu numeru i opuszczeniu kontrolki, po wciśnięciu przycisku OK itd.) Wymagane zrzuty ekranu:

- Aplikacja konsolowa – dokumentująca interakcje dowolna liczba zrzutów o nazwach *konsola1*, *konsola2*, ...
- Aplikacja desktopowa – dokumentująca interakcje dowolna liczba zrzutów o nazwach *desktop1*, *desktop2*, ...

W edytorze tekstu pakietu biurowego utwórz plik z dokumentacją i nazwij go *egzamin*. Dokument powinien zawierać podpisane zrzuty ekranu oraz zapisane informacje dotyczące narzędzi, z których korzystał zdający podczas egzaminu:

- Nazwę systemu operacyjnego
- Nazwy środowisk programistycznych
- Nazwy języków programowania

Zrzuty ekranu i dokument umieść w podfolderze *dokumentacja*.

**UWAGA:** Nagraj płytę z rezultatami pracy. W folderze z numerem zdającego powinny się znajdować podfoldery *desktopowa*, *dokumentacja*, *konsola*. W folderze *desktopowa*: spakowany cały projekt aplikacji desktopowej, pliki ze źródłami interfejsu i logiki, opcjonalnie plik uruchomieniowy. W folderze *dokumentacja*: pliki ze zrzutami oraz plik *egzamin*. W folderze *konsola*: spakowany cały projekt aplikacji konsolowej, pliki źródłowe, opcjonalnie plik uruchomieniowy. Opisz płytę swoim numerem zdającego i pozostaw na stanowisku, zapakowaną w pudełku wraz z arkuszem egzaminacyjnym.

**Czas przeznaczony na wykonanie zadania wynosi 180 minut.**

**Ocenie będą podlegać 4 rezultaty**

- implementacja, kompilacja, uruchomienie programu,
- aplikacja konsolowa,
- aplikacja desktopowa,
- dokumentacja aplikacji.