

Nazwa kwalifikacji: **Projektowanie, programowanie i testowanie aplikacji**
Oznaczenie kwalifikacji: **INF.04**
Numer zadania: **01**
Wersja arkusza: **SG**

Wypełnia zdający

Numer PESEL zdającego*

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Miejsce na naklejkę z numerem
PESEL i z kodem ośrodka

Czas trwania egzaminu: **180** minut.

INF.04-01-23.01-SG

EGZAMIN ZAWODOWY

Rok 2023

CZĘŚĆ PRAKTYCZNA

**PODSTAWA PROGRAMOWA
2019**

Instrukcja dla zdającego

1. Na pierwszej stronie arkusza egzaminacyjnego wpisz w oznaczonym miejscu swój numer PESEL i naklej naklejkę z numerem PESEL i z kodem ośrodka.
2. Na KARCIE OCENY w oznaczonym miejscu przyklej naklejkę z numerem PESEL oraz wpisz:
 - swój numer PESEL*,
 - oznaczenie kwalifikacji,
 - numer zadania,
 - numer stanowiska.
3. Sprawdź, czy arkusz egzaminacyjny zawiera 5 stron i nie zawiera błędów. Ewentualny brak stron lub inne usterki zgłoś przez podniesienie ręki przewodniczącemu zespołu nadzorującego.
4. Zapoznaj się z treścią zadania oraz stanowiskiem egzaminacyjnym. Masz na to 10 minut. Czas ten nie jest wliczony do czasu trwania egzaminu.
5. Czas rozpoczęcia i zakończenia pracy zapisze w widocznym miejscu przewodniczący zespołu nadzorującego.
6. Wykonaj samodzielnie zadanie egzaminacyjne. Przestrzegaj zasad bezpieczeństwa i organizacji pracy.
7. Po zakończeniu wykonania zadania pozostaw arkusz egzaminacyjny z rezultatami oraz KARTĘ OCENY na swoim stanowisku lub w miejscu wskazanym przez przewodniczącego zespołu nadzorującego.
8. Po uzyskaniu zgody zespołu nadzorującego możesz opuścić salę/miejsce przeprowadzania egzaminu.

Powodzenia!

* w przypadku braku numeru PESEL – seria i numer paszportu lub innego dokumentu potwierdzającego tożsamość

Zadanie egzaminacyjne

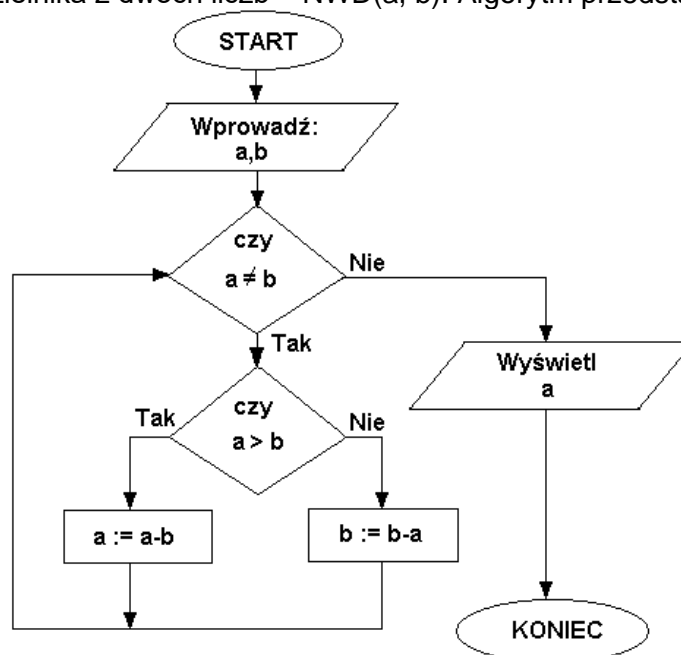
UWAGA: katalog z rezultatami pracy oraz płytę należy opisać numerem zdającego, którym został podpisany arkusz, czyli numerem PESEL lub w przypadku jego braku numerem paszportu. Dalej w zadaniu numer ten jest nazwany numerem zdającego.

Wykonaj aplikację konsolową oraz desktopową według wskazań. Wykonaj dokumentację do aplikacji konsolowej, zgodnie z opisem w części III instrukcji do zadania. 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 algorytm Euklidesa do szukania największego wspólnego dzielnika z dwóch liczb – NWD(a , b). Algorytm przedstawiono na schemacie.



Algorytm Euklidesa

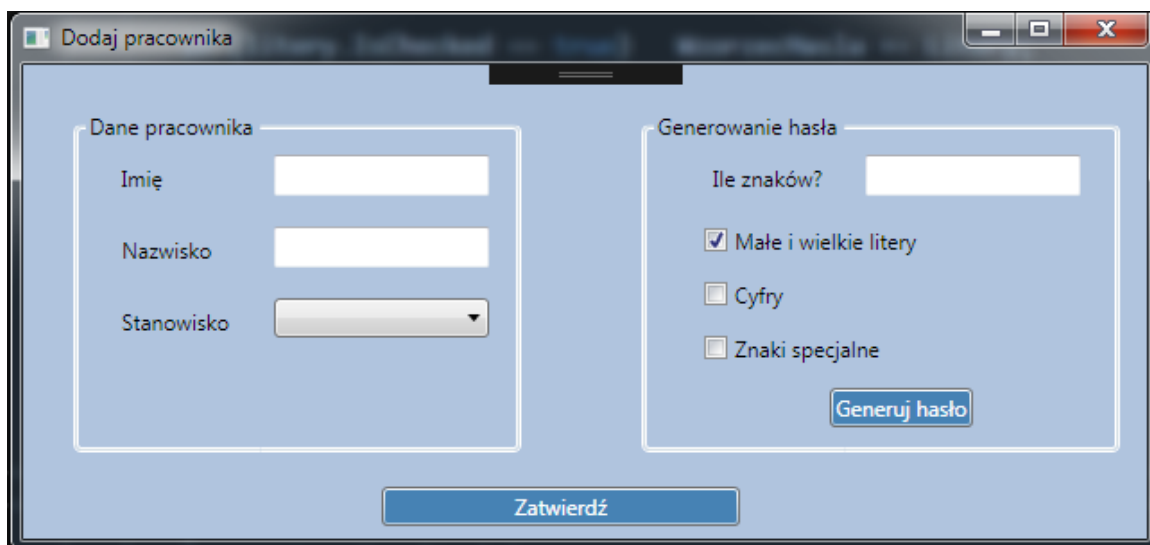
Założenia aplikacji:

- Obiektowy język programowania zgodny z zainstalowanym na stanowisku egzaminacyjnym: C++ lub C#, lub Java, lub Python
- Implementacja algorytmu w pełni zgodna z przedstawionym na schemacie algorytmem
- Liczby a i b należą do zbioru liczb całkowitych dodatnich (odpowiedni typ lub kontrola poprawności wpisanej liczby)
- Szukanie NWD zaimplementowane w funkcji o dwóch argumentach i zwracanej wartości największego wspólnego dzielnika. Funkcja nie może zawierać operacji wejścia - wyjścia
- Program główny testuje działanie funkcji i zawiera operacje wejścia - wyjścia
- Program powinien być zapisany czytelnie, z zachowaniem zasad czystego formatowania kodu, należy stosować nazwy zmiennych zgodne z nazwami zastosowanymi na schemacie blokowym, pozostałe nazwy muszą być znaczące
- Do kodu należy dołączyć dokumentację, która została opisana w części III zadania egzaminacyjnego.

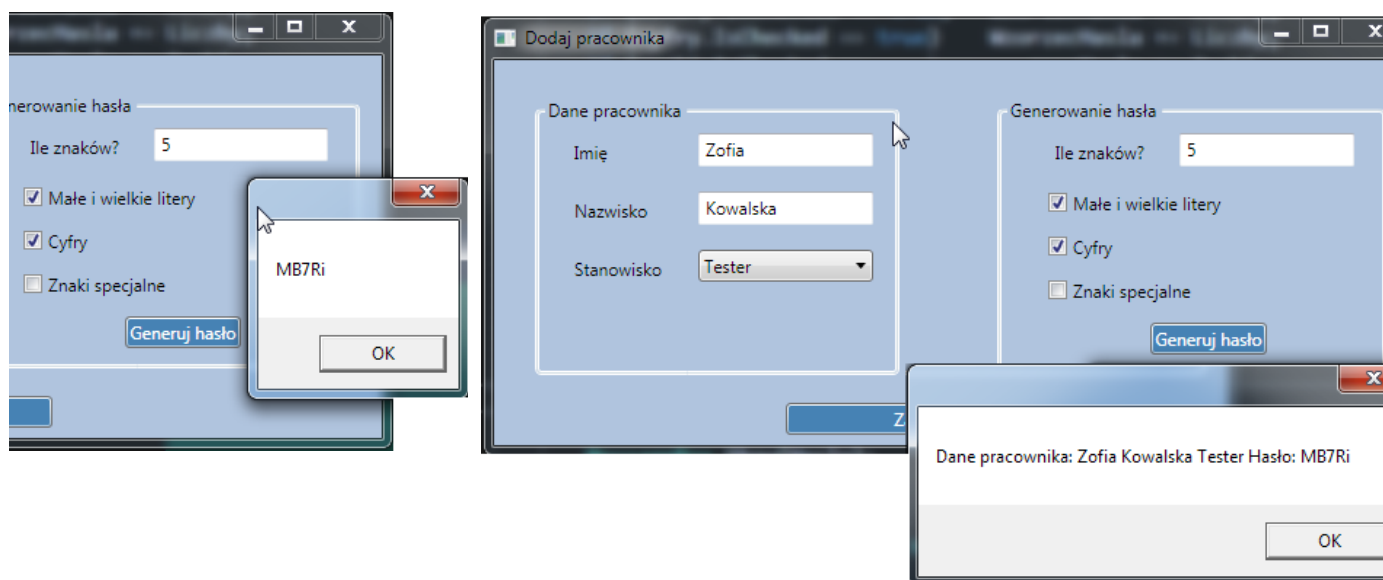
Kod aplikacji przygotuj do nagrania na płytę. W folderze *konsola* powinno znaleźć się archiwum całego projektu o nazwie *konsola.zip*, plik z kodem źródłowym programu oraz plik uruchomieniowy, jeżeli istnieje.

Część II. Aplikacja desktopowa

Za pomocą dostępnego na stanowisku egzaminacyjnym środowiska programistycznego wykonaj aplikację desktopową do wprowadzania danych pracownika z możliwością generowania hasła.



Obraz 1. Stan początkowy aplikacji



Obraz 2. Fragment okna po wybraniu przycisku „Generuj hasło”

Obraz 3. Po wybraniu przycisku „Zatwierdź”

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

Opis wyglądu aplikacji

- Okno o nazwie „Dodaj pracownika”. W nazwie okna należy wstawić także numer zdającego
- Kontrolki rozmieszczone zgodnie z obrazem 1
- Kontrolki w grupie „Dane Pracownika”:
 - pole edycyjne poprzedzone etykietą o treści „Imię”
 - pole edycyjne poprzedzone etykietą o treści „Nazwisko”
 - lista rozwijalna podpisana „Stanowisko” z elementami: Kierownik, Starszy programista, Młodszy programista, Tester

- Kontrolki w grupie „Generowanie hasła”:
 - pole edycyjne poprzedzone etykietą o treści „Ile znaków?”
 - trzy pola wyboru podpisane: „Małe i wielkie litery”, „Cyfry”, „Znaki specjalne”. Pierwsze pole jest domyślnie zaznaczone
 - przycisk o treści „Generuj hasło”
- Na dole okna znajduje się przycisk „Zatwierdź”, jest on wyraźnie dłuższy niż przycisk do generowania hasła
- Okno ma tło koloru LightSteelBlue (#B0C4DE)
- Przyciski mają tło koloru SteelBlue (#4682B4) i biały kolor czcionki

Działanie aplikacji po wybraniu przycisku „Generuj hasło”:

- Generowane jest hasło o liczbie znaków określonej w polu edycyjnym
- Poszczególne znaki hasła są wybierane losowo z zestawu małych liter
- Jeżeli zaznaczono checkbox „Małe i wielkie litery”, jeden ze znaków jest losowany z zestawu wielkich liter
- Jeżeli zaznaczono checkbox „Cyfry”, jeden ze znaków hasła jest losowany z zestawu cyfr
- Jeżeli zaznaczono checkbox „Znaki specjalne”, jeden ze znaków hasła jest losowany z zestawu znaków specjalnych
- Dla uproszczenia zadania można przyjąć zawsze określony znak (np. pierwszy), który jest losowany z cyfr. Podobnie należy postąpić ze znakami specjalnymi (np. drugi znak)
- Hasło jest wypisywane jako komunikat zgodnie z obrazem 2 po wybraniu przycisku „Generuj hasło”
- Za zestaw liter przyjmuje się wszystkie litery z klawiatury małe i wielkie alfabetu łacińskiego
- Za zestaw cyfr przyjmuje się kolejne cyfry od 0 do 9
- Za zestaw znaków specjalnych przyjmuje się znaki !@#\$%^&*()_+==
- Hasło oraz zestawy znaków są przechowywane w zmiennych typu napisowego

Działanie aplikacji po wybraniu przycisku „Zatwierdź”:

- Wyświetlany jest komunikat z wypełnionymi danymi pracownika oraz wygenerowanym wcześniej hasłem, zgodnie z obrazem 3

UWAGA: dla uproszczenia aplikacji nie trzeba sprawdzać czy pole edycyjne jest wypełnione i czy liczba znaków w hasle jest wystarczająca dla wstawienia małych, wielkich liter, liczb i znaków specjalnych

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.

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

Część III. Dokumentacja utworzonych aplikacji

Wykonaj dokumentację do aplikacji utworzonych na egzaminie. W kodzie źródłowym aplikacji konsolowej utwórz nagłówek funkcji wyznaczającej NWD, według wzoru. Nagłówek powinien znaleźć się nad definicją funkcji. W miejscu nawiasów <> należy podać odpowiednie opisy. W miejscu autor należy podać numer zdającego.

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 (liczba gwiazdek dowolna większa od 5)

```
*****
nazwa funkcji:      <nazwa>
opis funkcji:      <krótki opis co robi funkcja>
parametry:         <nazwa parametru 1 - znaczenie parametru
                   nazwa parametru 2 - znaczenie parametru>
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. Wykonać należy co najmniej tyle zrzutów, ile interakcji podejmuje aplikacja (np. stan początkowy, po wypisaniu hasła, po wyświetleniu danych itd.) Wymagane zrzuty ekranu:

- Aplikacja konsolowa – zrzuty nazwane: *konsola1*, *konsola2*, ...
- Aplikacja desktopowa – zrzuty nazwane: *desktop1*, *desktop2*, ...

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

- Nazwę systemu operacyjnego, na którym pracował zdający,
- Nazwy środowisk programistycznych, z których zdający korzystał na egzaminie,
- Nazwy języków programowania,
- Opcjonalnie komentarz do wykonanej pracy.

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 *dokumentacja*: pliki ze zrzutami oraz plik *egzamin*. W folderze *konsola*: spakowany cały projekt aplikacji konsolowej, źródło, opcjonalnie plik uruchomieniowy. W folderze *desktopowa*: spakowany cały projekt aplikacji desktopowej, pliki ze źródłami interfejsu i logiki, opcjonalnie plik uruchomieniowy. Opisz płytę 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.

