

Nazwa kwalifikacji: **Projektowanie, programowanie i testowanie aplikacji**

Oznaczenie kwalifikacji: **INF.04**

Numer zadania: **01**

Kod arkusza: **INF.04-01-22.06-SG**

Wersja arkusza: **SG**

Lp.	Elementy podlegające ocenie/kryteria oceny
R.1	Rezultat 1: Implementacja, kompilacja, uruchomienie programu
	<i>Uwaga: kryteria należy odnieść do aplikacji konsolowej, jeżeli ta nie istnieje, zastosować 1.1 ÷ 1.6 do aplikacji mobilnej Wystarczy, że sprawdzaną cechę zastosowano dla większości przypadków w kodzie</i>
R.1.1	Kod źródłowy zapisano w sposób czytelny: instrukcje w osobnych liniach, stosowane spacje pomiędzy operatorami, konsekwentnie stosowana wybrana konwencja dla nawiasów klamrowych instrukcji blokowej
R.1.2	Kod zapisano z wcięciami dla zagłębień bloków
R.1.3	Użyto znaczące nazewnictwo funkcji
R.1.4	Użyto znaczące nazewnictwo zmiennych (przynajmniej jedna). Wyjątkami od reguły są zmienne bufor, tmp, iteratory pętli itp. Dopuszcza się x jako nazwę szukanej. Kryterium nie jest spełnione tylko wtedy, gdy nazwy zmiennych nic nie znaczą, np. a, fun, foo
R.1.5	Zastosowano typy zmiennych pasujące do problemu (np. dowolny typ całkowity dla tablicy; typ całkowity dla iteratora pętli) oraz typ elementu tablicy jest taki sam jak typ szukanej. W przypadku Python zastosowano jawną konwersję do typu int dla wczytywanej liczby oraz zastosowano funkcję losującą liczby całkowite (randint)
R.1.6	Podjęto próbę skompilowania kodu, co udokumentowano zrzutem ekranowym przedstawiającym wykonywany program lub jego kompilację
R.1.7	Program nawiązuje zrozumiałą komunikację z użytkownikiem. Wyświetla przynajmniej jeden komunikat: Monit o podanie szukanej liczby lub komunikat z odnalezionym indeksem elementu lub jego brakiem. Jeżeli kod nie uruchamia się z powodu błędów kompilacji - sprawdzić w kodzie aplikacji
R.2	Rezultat 2: Aplikacja konsolowa
	<i>Uwaga: Kryteria 2.1 - 2.3, 2.6, 2.7 należy sprawdzić w kodzie programu, sprawdzane elementy muszą być zapisane zgodnie ze składnią Gdy aplikacja nie uruchamia się, a zdający zapisał zrzut ekranu z uruchomienia aplikacji należy sprawdzić powód braku kompilacji. Jeśli występują błędy w plikach źródłowych zdającego kryteria 2.4, 2.5, 2.8 nie są spełnione. Jeżeli błędy występują w innych plikach ocenić na podstawie kodu i zrzutu ekranu</i>
R.2.1	Program składa się z programu głównego oraz dwóch funkcji: do wypełniania tablicy i do szukania elementu w tablicy. Zawartość przynajmniej jednej funkcji jest zgodna z przeznaczeniem (może być niedokończona lub zawierać błędy)
R.2.2	Utworzono funkcję szukającą, która zwraca wartość indeksu (typ liczbowy). Typ funkcji jest zgodny z typem zwracanej wartości
R.2.3	Zastosowano algorytm przeszukiwania z wartownikiem: wartość szukana jest wstawiona do tablicy za ostatnim jej elementem, pętla przeszukująca działa od pierwszego elementu tablicy do momentu odnalezienia wartości
R.2.4	Tylko gdy wyszukany indeks jest ostatnim w tablicy, wyświetlany jest komunikat o nieodnalezieniu elementu
R.2.5	Tylko gdy wyszukany indeks nie jest ostatnim w tablicy wyświetlany jest indeks wyszukanego elementu

R.2.6	Tablica jest wypełniana liczbami losowymi z zakresu od 1 do 100 oraz liczba szukana jest pobierana z klawiatury
R.2.7	Elementy tablicy, oddzielone przecinkami, są wyświetlane na ekranie
R.2.8	Program nie powoduje błędów dostępu do pamięci lub wyjątków: podczas wstawiania elementu szukanego tablicy oraz podczas przeszukiwania w pętli
R.2.9	Program kompiluje się i uruchamia w konsoli, co udokumentowano zrzutem ekranu
R.3	Rezultat 3: Aplikacja mobilna
	<i>Uwaga: jeżeli jest to możliwe uruchomić aplikację na tym samym urządzeniu, na którym uruchamiał zdający. Należy uwzględnić różnice pomiędzy emulacjami - takie cechy jak marginesy, wielkości bloków itp. nie należy brać pod uwagę. Kryteria 3.1 ÷ 3.4, 3.7, 3.8 sprawdzić w kodzie źródłowym, sprawdzane elementy muszą być zapisane zgodnie ze składnią. Gdy aplikacja nie uruchamia się, a zdający zapisał zrzuty ekranu z uruchomienia aplikacji należy sprawdzić powód braku kompilacji. Jeśli występują błędy w plikach źródłowych zdającego kryteria 3.5, 3.6, 3.9, 3.10 nie są spełnione. Jeżeli błędy występują w innych plikach lub bibliotekach sprawdzić w kodzie oraz na zrzucie ekranu</i>
R.3.1	Zastosowano język znaczników XML/XAML lub inny do opisu interfejsu użytkownika oraz kod zawiera przynajmniej jeden element / kontrolkę interfejsu graficznego
R.3.2	Zastosowano rozkład typu liniowego wertykalnego (LinearLayout / StackLayout lub inny o tej idei) z zagłębionym rozkładem liniowym horyzontalnym dla przycisków oraz ustawiono margines wewnętrzny górny dla całej strony lub rozkładu 20px (lub dp)
R.3.3	Zastosowano kontrolki tekstu i przycisku. Tytuł jest zapisany czcionką o największym rozmiarze spośród stosowanych w aplikacji, czcionka napisu "Opis" jest pogrubiona. Napis z liczbą polubień jest wyrównany do prawej
R.3.4	Nadano kolor Teal (#008080) dla tła przycisków i rozkładu, w którym się one znajdują, kolor czcionki przycisków jest biały, kolor tekstu "Odwiedź..." jest Gray (#808080)
R.3.5	Aplikacja wyświetla obraz o nazwie <i>obraz.jpg</i> , jest on rozciągnięty na całą szerokość ekranu (może być obcięty) np. AspectFill, FitCenter, adjustViewBounds
R.3.6	Wyświetlono linię horyzontalną (np. BoxView o wysokości 1) koloru Gray (#808080)
R.3.7	Zdefiniowano zmienną typu liczbowego całkowitego, która na początku działania aplikacji jest równa 0 oraz jest inkrementowana i dekrementowana w funkcjach obsługujących zdarzenia kliknięcia przycisku (wystarczy jedna zdefiniowana funkcja)
R.3.8	Gdy licznik polubień jest równy 0 wciśnięcie przycisku "USUŃ" nie zmniejsza jego wartości
R.3.9	Po wciśnięciu przycisku "POLUB" wyświetla się zwiększona o 1 liczba polubień oraz po wciśnięciu przycisku "USUŃ" wyświetla się zmniejszona o 1 liczba polubień
R.3.10	Aplikacja kompiluje się i uruchamia w emulatorze. Zawiera co najmniej 7 elementów oraz wygląd zgodny z obrazem 1a lub 1b w arkuszu egzaminacyjnym. Napis liczby polubień wyświetla "0 polubień", co udokumentowano zrzutem ekranu
R.4	Rezultat 4: Dokumentacja aplikacji
	<i>Uwaga: Komentarz z kryteriów 4.1 ÷ 4.5 musi być zgodny ze stanem faktycznym z kodu źródłowego, nawet jeżeli w kodzie są błędy logiczne (liczba pól, typy). Zrzuty ekranu z kryteriów 4.6 i 4.7 muszą zawierać cały obszar ekranu z widocznym paskiem zadań. Dokumentacja z kryterium 4.8 zapisana jest w pliku egzamin</i>
R.4.1	Dla funkcji wyszukiwającej z aplikacji konsolowej zapisano nagłówek w postaci komentarza zgodny z Listingiem 1 z arkusza egzaminacyjnego, umieszczony w komentarzu wieloliniowym lub kilku jednoliniowych
R.4.2	W komentarzu ujęto nazwę funkcji
R.4.3	Jeżeli funkcja przyjmuje argumenty w komentarzu ujęto ich nazwy wraz z opisem oraz ujęto typ zwracany (nazwa lub opis np. "liczba całkowita") z krótką informacją

R.4.4	W komentarzu zawarto informacje o funkcji, które zawierają przynajmniej sformułowanie dotyczące algorytmu i metody, np. "przeszukiwanie tablicy z wartownikiem"
R.4.5	W komentarzu ujęto numer zdającego
R.4.6	Zapisano przynajmniej jeden zrzut ekranu z uruchomienia lub kompilacji aplikacji konsolowej, na zrzucie widoczne jest środowisko, w którym powstała aplikacja
R.4.7	Zapisano przynajmniej jeden zrzut ekranu z uruchomienia lub kompilacji aplikacji mobilnej, na zrzucie widoczne jest środowisko, w którym powstała aplikacja
R.4.8	Dokumentacja zawiera: nazwę systemu operacyjnego, nazwy środowisk, nazwy języków programowania, nazwę emulatora systemu mobilnego