

Nazwa kwalifikacji: **Projektowanie, programowanie i testowanie aplikacji**
Symbol 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.04-01-24.06-SG

EGZAMIN ZAWODOWY

Rok 2024

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 7 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 wliczany 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 mobilną według wskazań. Wykonaj dokumentację 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: *konsolowa*, *mobilna*, *dokumentacja*. Po wykonaniu każdej aplikacji, jej pełny kod (cały folder projektu) **spakuj do archiwum**. Następnie pozostaw w podfolderze jedynie spakowane archiwum, skopiowane z projektu pliki źródłowe, których treść była modyfikowana oraz, jeżeli istnieje, plik wykonywalny.

Część I. Aplikacja konsolowa

Napisz program implementujący logikę gry w kości. Do gry stosowane są sześciennie kostki z oczkami od 1 do 6. Działanie programu przedstawia obraz 1.

Działanie programu:

1. Pobiera z klawiatury liczbę kostek do rzucenia z przedziału od 3 do 10. Punkt 1 jest powtarzany dopóki podana liczba nie należy do przedziału $<3, 10>$ - obraz 2.
2. Losuje tyle liczb całkowitych z przedziału od 1 do 6, ile podano z klawiatury, co odpowiada rzutom kośćmi
3. Wypisuje ile wyrzucono oczek na każdej kostce w formacie „Kostka <numer-kostki>: <liczba-oczek>”
4. Liczy punkty i je wyświetla zgodnie z obrazem 1.
5. Pyta grającego czy powtórzyć grę, gdy podano:
 - ‘t’: idź do punktu 2
 - ‘n’: skończ działanie programu.

Zasada liczenia punktów za rzuty kostką:

- Punkty są sumą oczek, gdy dana liczba została wylosowana dwa razy lub więcej.
- Przykład 1 z obrazu 1: dla wylosowanych oczek: 1, 5, 2, 2, 5 uzyskano 14 punktów. Liczba 1 jest pomijana, została tylko raz wylosowana. Liczby 2 oraz 5 zostały dwa razy wylosowane zatem suma $2 + 2 + 5 + 5$ wynosi 14 punktów.
- Przykład 2: wylosowano 2, 1, 2, 3, 2. Oczka 1, 3 zostały wylosowane tylko raz, więc są pomijane, 2 zostało wylosowane trzy razy, zatem $2 + 2 + 2$ jest równe 6 punktom.

```
Ile kostek chcesz rzucić?(3 - 10)
5
Kostka 1: 1
Kostka 2: 5
Kostka 3: 2
Kostka 4: 2
Kostka 5: 5
Liczba uzyskanych punktów: 14
Jeszcze raz? (t/n)
t
Kostka 1: 2
Kostka 2: 1
Kostka 3: 2
Kostka 4: 3
Kostka 5: 4
Liczba uzyskanych punktów: 4
Jeszcze raz? (t/n)
t
Kostka 1: 2
Kostka 2: 2
Kostka 3: 3
Kostka 4: 1
Kostka 5: 3
Liczba uzyskanych punktów: 10
Jeszcze raz? (t/n)
n
```

Obraz 1. Działanie aplikacji konsolowej

Założenia do programu:

- Program wykonywany w konsoli
- Zastosowany obiektowy język programowania zgodny z zainstalowanym na stanowisku egzaminacyjnym: C++ lub C#, lub Java, lub Python
- Dowolne podejście: strukturalne lub obiektowe
- Wynik pojedynczego losowania kostek jest zapisywany do tablicy lub kolekcji. Dla przykładu 1 z obrazu 1 do tablicy zostały zapisane liczby 1, 5, 2, 2, 5
- Program zawiera funkcję / metodę losującą liczby z przedziału od 1 do 6 (rzucającą wszystkimi kośćmi). W przypadku podejścia strukturalnego funkcja nie może korzystać ze zmiennych globalnych
- Program zawiera funkcję / metodę liczącą punkty. Suma punktów jest zwracana z funkcji / metody. W przypadku podejścia strukturalnego funkcja nie może korzystać ze zmiennych globalnych
- Program powinien podejmować zrozumiałą komunikację z użytkownikiem
- W programie może być zastosowane angielskie lub polskie nazewnictwo zmiennych i funkcji / metod
- Program powinien być zapisany czytelnie, z zachowaniem zasad czystego formatowania kodu, należy stosować znaczące nazwy zmiennych, funkcji / metod
- Dokumentację aplikacji należy utworzyć zgodnie z opisem w części III treści zadania.

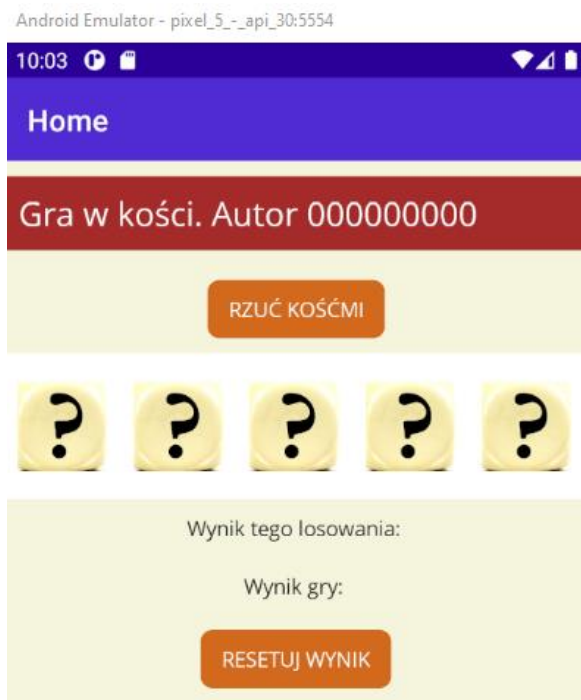
```
Ile kostek chcesz rzucić?(3 - 10)
2
Ile kostek chcesz rzucić?(3 - 10)
1
Ile kostek chcesz rzucić?(3 - 10)
11
Ile kostek chcesz rzucić?(3 - 10)
4
Kostka 1: 3
Kostka 2: 4
Kostka 3: 2
Kostka 4: 3
Liczba uzyskanych punktów: 6
Jeszcze raz? (t/n)
```

Obraz 2. Niepoprawna i poprawna liczba kostek

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

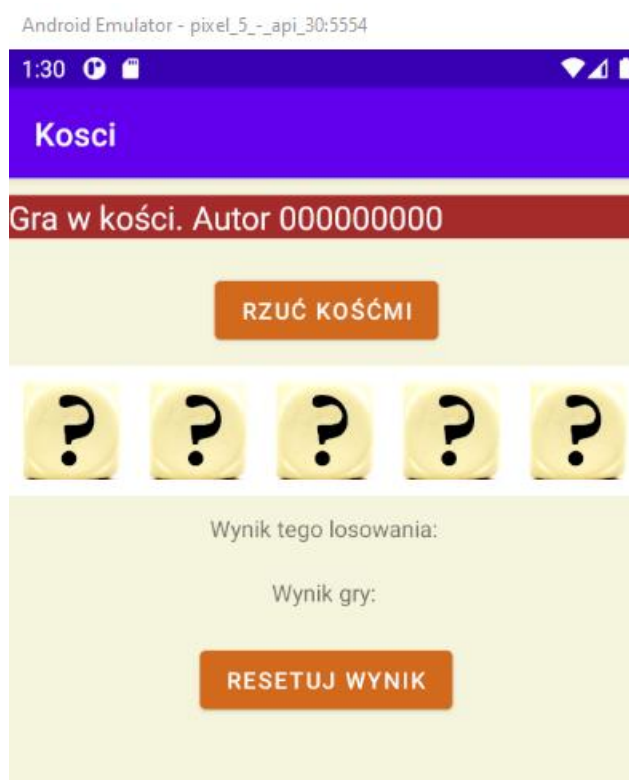
Część II. Aplikacja mobilna

Za pomocą środowiska programistycznego dostępnego na stanowisku egzaminacyjnym wykonaj aplikację mobilną do gry w kości. Do zbudowania aplikacji zastosuj obrazy z archiwum *pliki1.zip* zabezpieczonego hasłem **Gra^Kosci5**



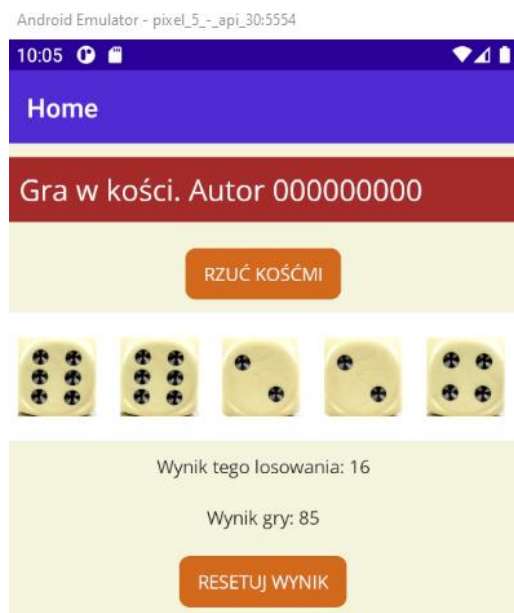
Obraz 3.

Środowisko .NET MAUI, stan początkowy.
Emulacja Pixel 5



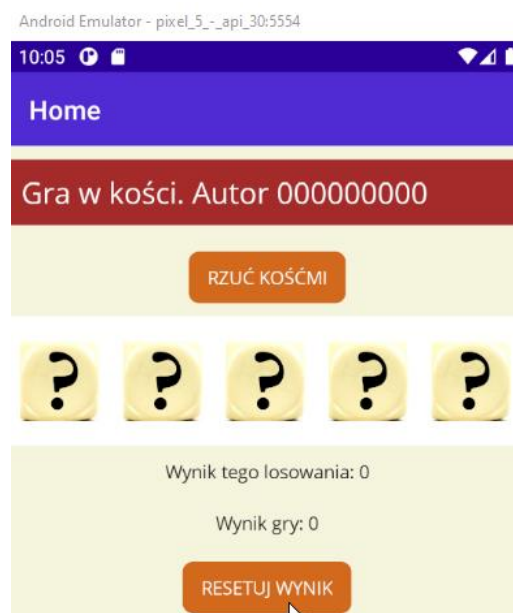
Obraz 4.

Środowisko Android Studio, stan początkowy.
Emulacja Pixel 5



Obraz 5.

Po wciśnięciu przycisku RZUĆ KOŚĆMI



Obraz 6.

Po wciśnięciu przycisku
RESETUJ WYNIK

Stan początkowy aplikacji został przedstawiony na obrazach 3 i 4. Interakcje w aplikacji zostały przedstawione na obrazach 5 i 6. W zależności od zastosowanego środowiska programistycznego oraz emulowanego systemu widok może nieznacznie różnić się od przedstawionego.

Elementy aplikacji:

- Napis (etykieta) z tytułem gry: „Gra w kości. Autor ”, dalej wstawiony numer zdającego
- Przycisk o treści „RZUĆ KOŚĆMI”
- Pięć obrazów w stanie początkowym wypełnionych grafiką *question.jpg*
- Napis „Wynik tego losowania: ”
- Napis „Wynik gry: ”
- Przycisk o treści „RESETUJ WYNIK”
- Rozmieszczenie elementów zgodne z obrazami 3 i 4

Działanie aplikacji:

- Do wykonania aplikacji można wykorzystać fragmenty kodu z aplikacji konsolowej. Dla uproszczenia aplikacji, gra odbywa się zawsze na pięciu kościach
- Po wciśnięciu przycisku „RZUĆ KOŚĆMI” (obraz 5):
 - Losowanych jest 5 liczb z zakresu od 1 do 6, które są wyświetlone w postaci pięciu obrazów kości
 - Obliczana jest suma punktów dla rzutu kośćmi zgodnie z algorytmem podanym w opisie aplikacji konsolowej oraz wynik wyświetlany jest w napisie z wynikiem losowania
 - Wynik gry jest powiększany o obliczony wynik rzutu i wyświetlany w napisie z wynikiem gry
- Po wciśnięciu przycisku „RESETUJ WYNIK” (obraz 6):
 - Pięć obrazów kości wypełnione zostaje grafiką *question.jpg*
 - Wynik całej gry jest resetowany do wartości 0
 - W napisach dotyczących wyniku losowania i wyniku gry jest zapisana wartość 0

Założenia aplikacji:

- Interfejs użytkownika zapisany za pomocą języka znaczników wspieranego w danym środowisku (np. XAML, XML)
- Zastosowany rozkład liniowy wertykalny (Linear / Stack lub inny o tej idei), z zagłębionym rozkładem liniowym horyzontalnym dla pięciu obrazów
- Kolory tła
 - Rozkład wertykalny lub strona: Beige (#F5F5DC)
 - Napis z tytułem strony: Brown (#A52A2A)
 - Oba przyciski: Chocolate (#D2691E)
 - Rozkład horyzontalny: biały
- Biały kolor czcionki dla napisu z tytułem gry
- Rozmiar i marginesy dla pięciu obrazów należy dobrać tak, aby wszystkie obrazy były widoczne w całości, nie stykały się ze sobą oraz wypełniały całą szerokość rozkładu. Dla emulacji Pixel5 z obrazu 3 zastosowano wysokość obrazu 60 i marginesy 9
- Marginesy dla pozostałych kontrolki i elementów strony należy dobrać tak, aby kontrolki nie przylegały do siebie w pionie i w poziomie. Dla emulacji Pixel5 z obrazu 3 zastosowano marginesy górny i dolny 10
- Rozmiar czcionki napisu z tytułem gry jest największy względem innych napisów
- Napisy z wynikami oraz przyciski są wyśrodkowane, napis z tytułem oraz rozkład horyzontalny wypełniają w poziomie całą szerokość strony
- Aplikacja powinna być zapisana czytelnie, z zachowaniem zasad czystego formatowania kodu, należy stosować znaczące nazwy zmiennych i funkcji.

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

Kod aplikacji przygotuj do nagrania na płytę. W podfolderze *mobilna* powinno znaleźć się archiwum całego projektu o nazwie *mobilna.zip*, skopiowane z projektu plik (lub pliki) z kodem źródłowym interfejsu użytkownika (XAML lub XML) oraz plik źródłowy kodu skojarzonego z interfejsem użytkownika.

Część III. Dokumentacja aplikacji konsolowej

Wykonaj dokumentację do aplikacji utworzonych na egzaminie. W kodzie źródłowym aplikacji konsolowej za pomocą komentarza utwórz nagłówek wybranej funkcji (metody) według wzoru z listingu 1. Komentarz powinien znaleźć się nad lub pod nazwą funkcji. W miejscu nawiasów <> należy podać odpowiednie opisy.

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:          <tu wstaw nazwę funkcji / metody>
opis:          <co wykonuje funkcja / metoda?>
parametry:     <opis parametru1, lub „brak”>
               <opis parametru2>
               ...
zwracany typ i opis:<nazwa typu i opis co jest zwracane lub „brak”>
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 tyle zrzutów, ile interakcji podejmuje aplikacja.

Wymagane zrzuty ekranu:

- Aplikacja konsolowa – dowolna liczba zrzutów nazwanych *konsola1*, *konsola2*, ... (np. stan początkowy, błędnie i poprawnie podana liczba kostek, kilka losowań, opcje 't' oraz na końcu opcja 'n')
- Aplikacja mobilna – dowolna liczba zrzutów nazwanych *mobile1*, *mobile2*, ... (np. stan początkowy, po wciśnięciu przycisku RZUĆ KOŚĆMI, po wciśnięciu przycisku „RESETUJ WYNIK”)

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

- Nazwę systemu operacyjnego, na którym pracował zdający
- Nazwy środowisk programistycznych, z których zdający korzystał na egzaminie
- Nazwę emulatora dla aplikacji mobilnej
- Nazwy użytych 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: *dokumentacja*, *konsolowa*, *mobilna*. W folderze *dokumentacja*: pliki ze zrzutami oraz plik *egzamin*. W folderze *konsolowa*: spakowany cały projekt aplikacji konsolowej, pliki źródłowe, opcjonalnie plik wykonywalny. W folderze *mobilna*: spakowany cały projekt aplikacji mobilnej, pliki z kodem źródłowym interfejsu i logiki. Po nagraniu płyty sprawdź poprawność nagrania. 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 mobilna,
- Dokumentacja aplikacji.

