

## Przeprowadzenie analizy obiektowej wymagań projektu

Projekt polega na stworzeniu gry Blackjack. Gracz w rozdawanych mu przez krupiera kartach uzyskać musi 21 punktów lub maksymalnie zbliżyć się do tej liczby, w żadnym wypadku nie mogąc jej przekroczyć. W blackjacku punktacja za poszczególne figury prezentuje się następująco:

- Blotki – zgodnie z wartością numerową, czyli na przykład ósemka oznacza 8 punktów
- Walet, Dama, Król – 10 punktów
- As – 1 lub 11 punktów, w zależności od tego, która opcja jest korzystniejsza dla gracza

Warunkiem wygranej jest uzyskanie większej liczby punktów od krupiera bez przebicia 21. Karty gracza może widzieć tylko on sam, z kolei jedna karta krupiera musi być otwarta. Gracz i krupier zaczynają rozgrywkę z danym zakładem. Na początku rozdawane są po dwie karty, po czym krupier musi dobrać tyle kart, żeby wartość jego ręki wynosiła minimum 17. Gracz może spasować lub podać opcje:

- stand – nie dobiera, stawia zakład
- hit – dobranie karty
- double – podwojenie stawki i dobranie maksymalnie jednej karty
- split – gdy gracz dostanie dwie karty o tej samej wartości, rozdziela je na dwie osobne ręce; każda ręka ma ten sam początkowy zakład i działa niezależnie od drugiej

Obiekty:

1. Karta
2. Talia
3. Reka
4. Osoba
5. Gracz
6. Kupier
7. Gra

Metody i atrybuty:

### 1. Karta

Atrybuty:

kolor - kolor karty

wartosc - wartość karty

Metody:

- podaj\_Wartosc() - zwraca punktację karty

## 2. Talia

Atrybuty:

karty - stos 52 kart bez asów

Metody:

- `tasuj()` - układa karty w losowej kolejności w stos
- `podaj_Karte()` - usuwa kartę ze stosu talii i przekazuje do gracza/krupiera

## 3. Ręka

Atrybuty:

karty - lista kart na ręce

Metody:

- `dobierz_Karte()` - ze stosu wyciągana jest karta
- `podaj_Wartosc()` - zwraca wartość ręki
- `blackjack()` - koniec gry, wygrana gracza, gracz zdobył równo 21 punktów
- `spalony()` - koniec gry, przegrana gracza, gracz zdobył ponad 21 punktów

## 4. Osoba (bazowa)

Atrybuty:

imie - nazwa gracza

ręka - obiekt klasy Ręka przechowujący karty osoby

Metody:

- `dobierz_Karte()` – dodaje kartę z talii do ręki

## 5. Gracz (dziedziczy po Osoba)

Atrybuty:

balans - saldo gracza

Metody:

- `stand()` – gracz kończy dobieranie kart
- `hit()` – gracz dobiera jedną kartę z talii
- `double()` – podwaja zakład, dobiera jedną kartę i wykonuje `stand()`
- `split()` – jeśli gracz ma dwie karty tej samej wartości, rozdziela je na dwie ręce i dobiera po jednej karcie do każdej
- `pass_()` – gracz poddaje się i traci połowę postawionego zakładu
- `postaw_Zaklad()` – ustawia wysokość zakładu, zmniejszając balans

## 6. Krupier (dziedziczy po Osoba)

Atrybuty:

brak dodatkowych atrybutów poza tymi z klasy bazowej

Metody:

- pokaz\_Karte() – pokazuje pierwszą kartę krupiera (widoczną dla gracza)
- dobierz\_Karte() – dobiera karty, za pierwszym razem aż wartość ręki osiągnie minimum 17

## 7. Gra

Atrybuty:

gracz

krupier

talía

obecny\_Zakład - wartość zakładu gracza

runda

Metody:

- rozpocznij() - rozpoczyna grę
- rozdaj\_Karty() - krupier rozdaje po 2 karty, obie karty gracza są widoczne dla niego, a krupier posiada 1 kartę widoczną i 1 zasłoniętą
- zakoncz() - gra zostaje zakończona
- porownaj() - porównywana jest wartość kart gracza i krupiera
- wypłac() - w przypadku zwycięstwa gracza jego balans zostaje zwiększony o podwojoną wartość zakładu

Diagram UML

