Tablice dwuwymiarowe — kółko i krzyżyk

Czas: 2 × 45 min | praca w parach (programista + tester, zamiana ról w połowie) | Wykład 1.07

Cel: Napiszesz grę w kółko i krzyżyk dla dwóch graczy w konsoli, przechowując planszę w tablicy dwuwymiarowej.

Po zajęciach potrafisz:

- deklaruje i wypełnia tablicę `char[,]`
- rysuje planszę pętlami zagnieżdżonymi
- sprawdza poprawność ruchu i warunki zwycięstwa
- dzieli program na funkcje lokalne

Sprzęt i materiały

- komputer z Windows 11 i dostępem do internetu
- pendrive lub konto w repozytorium do zapisania projektu

Oprogramowanie

- **Visual Studio Community 2026 (obciążenie „Programowanie aplikacji klasycznych platformy .NET")** — darmowe dla uczniów i nauczycieli; w pracowni zainstalowane visualstudio.microsoft.com/pl/downloads
- **.NET 10 SDK (LTS)** — instalowany razem z Visual Studio; sprawdzenie: `dotnet --version` dotnet.microsoft.com/pl-pl/download/dotnet/10.0

Bezpieczeństwo

- Zapisuj projekt w swoim folderze (np. `C:\Users\<login>\source\repos`), a na koniec zajęć skopiuj go na pendrive lub wyślij do repozytorium.
- Nie uruchamiaj programów i pakietów z nieznanych źródeł; pakiety NuGet instaluj tylko te, które wskazuje zadanie.

Przebieg zadania

1 Plansza

Utwórz projekt `KolkoKrzyzyk`.

```
    1   2   3
1   X | 0 |
   ---+---+---
2     | X |
   ---+---+---
3     |   | 0
```

1. Zadeklaruj `char[,] plansza = new char[3, 3];` i wypełnij ją spacjami.
2. Napisz funkcję `Rysuj()`, która wypisuje planszę z numerami wierszy i kolumn (wzór poniżej).
3. Sprawdź rysowanie, wpisując ręcznie kilka znaków do planszy.

2 Ruchy graczy

Gracze na zmianę podają wiersz i kolumnę, np. `2 3`.

1. Rozdziel tekst metodą `Split(' ', StringSplitOptions.RemoveEmptyEntries)` i zamień części na liczby.
2. Odrzuć ruch, gdy: dane nie są dwiema liczbami, liczby są spoza 1–3 albo pole jest zajęte. Pytaj ponownie.
3. Pamiętaj o przesunięciu indeksu: użytkownik podaje 1–3, tablica ma indeksy 0–2.
4. Po poprawnym ruchu zmień gracza: `gracz = gracz == 'X' ? 'O' : 'X';`.

3 Zwycięstwo i remis

Napisz funkcję `bool Wygral(char z)`.

1. Sprawdź w pętli trzy wiersze i trzy kolumny.
2. Sprawdź dwie przekątne.
3. Po każdym ruchu sprawdź zwycięstwo; po 9 ruchach bez zwycięzcy ogłoś remis.
4. Przetestuj z partnerem: wygraną w wierszu, w kolumnie, na każdej przekątnej, remis i wszystkie rodzaje błędnych ruchów.

4 Rozszerzenie (dla chętnych)

Wybierz jedno.

1. Kolorowe znaki: X na czerwono, O na niebiesko.
2. Licznik wygranych i ponowna gra bez zamykania programu.
3. Prosty przeciwnik komputerowy: losowe wolne pole.
4. Zatwierdź pracę w Git.

W sprawozdaniu umieść

- ☐ kod programu
- ☐ tabela testów wykonanych przez testera
- ☐ zrzut ekranu wygranej i remisu
- ☐ opis rozszerzenia (jeśli wykonane)

Pytania kontrolne

1. Czym różni się tablica `char[,]` od tablicy tablic `char[][]`?
2. Ile linii kodu trzeba by zmienić, aby plansza miała 4 × 4? Jak to uprościć?
3. Dlaczego sprawdzamy zwycięstwo tylko gracza, który właśnie wykonał ruch?
4. Jak program reaguje na pustą linię? Sprawdź.

Kryteria oceny

Kryterium	Punkty
Plansza i rysowanie	4
Ruchy z walidacją	5
Wykrywanie zwycięstwa i remisu	5
Testy w parach (tabela)	4
Rozszerzenie lub odpowiedzi na pytania	2
Razem	**20**

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).