

Aplikacja, którą stworzyliśmy jest naszą wariacją gry Pac-Man. Jednym z powodów była chęć zmierzenia się z implementacją kultowej gry, która w tym roku obchodzi swoje 40te urodziny. Staraliśmy się odwzorować jak najlepiej mechaniki przeciwników, co się udało. Informacje na temat gry i mechanik czerpaliśmy z:

- <https://en.wikipedia.org/wiki/Pac-Man>
- <https://gameinternals.com/understanding-pac-man-ghost-behavior>

Korzystaliśmy ze środowiska SFML 2.5.1

- <https://www.sfm-dev.org/index.php>

Zapewniło ono nam między innymi łatwą obsługę okien, zdarzeń oraz dostosowanie, animowanie i wyświetlanie grafik.

Program jest zupełnie darmowy, kod źródłowy jest dostępny na platformie github.com pod linkiem:

- https://github.com/marcciszek/prog_2_projekt_pacman

Sterowanie odbywa się za pomocą myszy (menu) i klawiatury (gra).

Mechanika przeciwników:

- Duch czerwony – Blinky:
 - Duch ten zaczyna pogoń za graczem tuż przed bazą
 - Jego celem jest pole, w którym aktualnie znajduje się Pac-Man
- Duch różowy - Pinky
 - Rozpoczyna podążanie za graczem na początku gry, ze środka bazy
 - Kieruje się na pole znajdujące się 4 kratki na wprost od gracza
- Duch niebieski – Inky
 - Ten przeciwnik włącza się do zabawy w momencie, gdy gracz zdobędzie 30 punktów
 - Miejsce, do którego się kieruje na pozór wydaje się losowe, dzieje się tak z dość ciekawego powodu, mierzona jest odległość od przeciwnika czerwonego do kratki, która znajduje się dwie kratki przed graczem co można zapisać jako wektor, podwojenie tego wektora wyznacza pozycję na którą się kieruje. Daje to ciekawy efekt, gdy duch czerwony jest blisko nas, duch niebieski też będzie w pobliżu
- Duch pomarańczowy - Clyde
 - Rozpoczyna pościg, gdy gracz zdobędzie 80 punktów
 - Ma mechanikę podobną do ducha czerwonego
 - Podąża za graczem, jeżeli jest w odległości większej niż 8 kratek od niego
 - Ucieka do lewego dolnego rogu mapy, gdy odległość ta jest mniejsza niż 8 kratek

Gra była testowana przez osoby nie związane ze światem programowania na różnorodnych maszynach. Opinie i spostrzeżenia innych użytkowników były o tyle cenne, że pokazały nam wcześniej niezauważone drobne błędy, które można było od razu naprawić.

Opis poszczególnych klas, zmiennych i metod są zawarte w kodzie źródłowym.