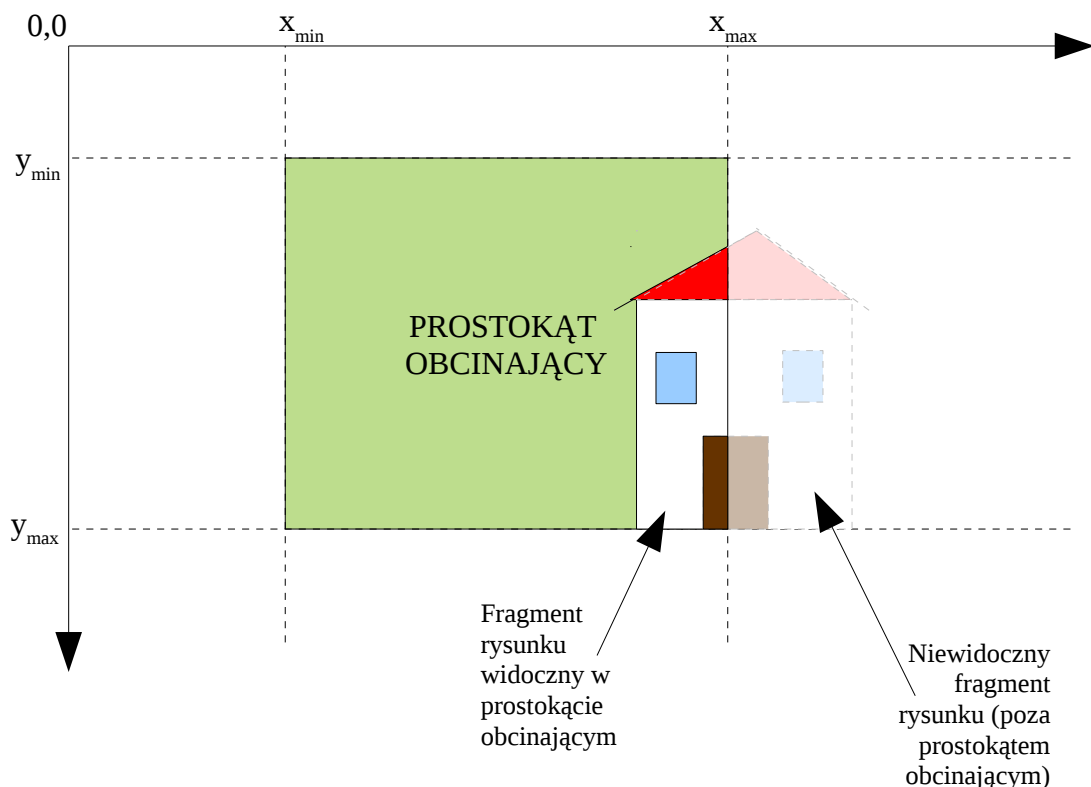


Instrukcja laboratoryjna 3	Grafika komputerowa 2D
	Temat: Obcinanie odcinków do prostokąta
	Przygotował: dr inż. Grzegorz Łukawski, mgr inż. Maciej Lasota, mgr inż. Tomasz Michno

1 Wstęp teoretyczny

1.1 Prostokąt obcinający

Prostokąt obcinający (nazywany również widokiem – ang. viewport) jest prostokątnym obszarem wyświetlającym wewnątrz siebie obraz, którego pozostała część jest niewidoczna poza jego obrębem. Najczęściej jego krawędzie są równoległe do krawędzi ekranu (choć czasami zdarzają się wyjątki, jednak na laboratoriach nie będziemy się nimi zajmowali).



Na rysunku można zauważyć prosty obraz, który jest widoczny jedynie częściowo, fragment pozostający poza prostokątem obcinającym nie zostaje wyświetlony.

Prostokąt obcinający można opisać poniższymi nierównościami:

$$\begin{aligned} x_{min} &\leq x \leq x_{max} \\ y_{min} &\leq y \leq y_{max} \end{aligned} \quad (1.1)$$

1.2 Obcinanie punktów

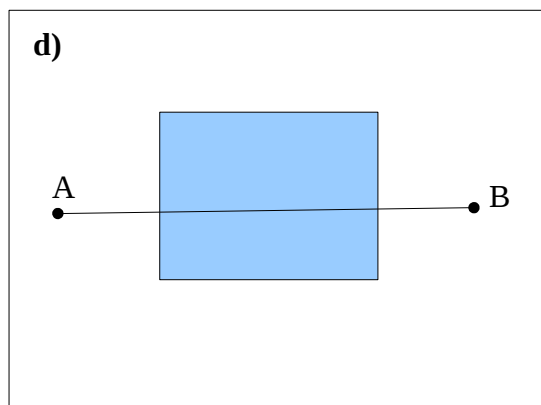
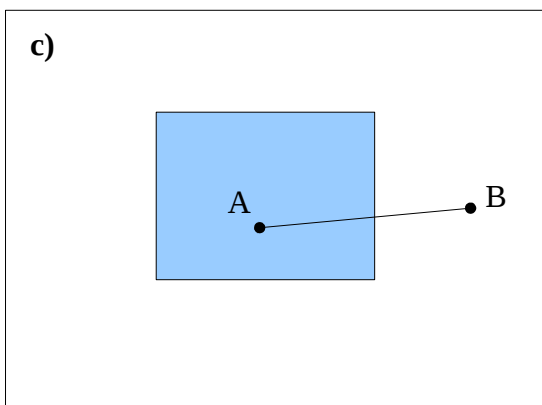
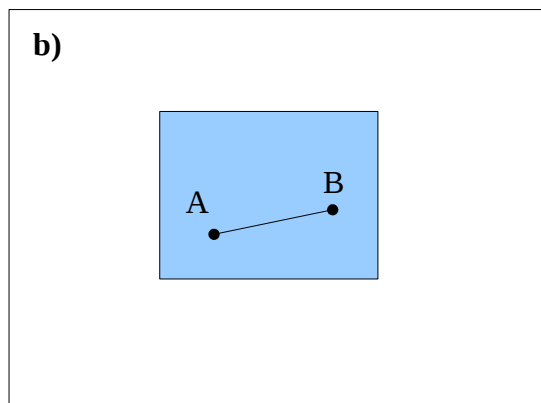
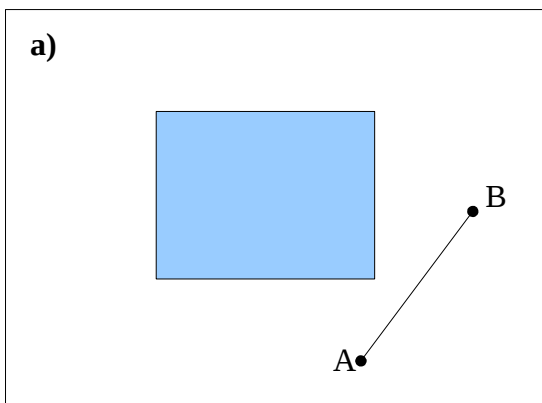
Sprawdzenie, czy punkt $P(x,y)$ leży wewnątrz prostokąta obcinającego sprowadza się do sprawdzenia nierówności (1.1). Jeżeli wszystkie nierówności zostają spełnione, punkt leży

wewnątrz, w przeciwnym przypadku na zewnątrz prostokąta.

1.3 Obcinanie odcinków

Obcinanie odcinków jest operacją trudniejszą niż obcinanie punktów, ponieważ może wystąpić jeden z czterech różnych przypadków:

- odcinek leży na zewnątrz prostokąta obcinającego
- odcinek leży wewnątrz prostokąta obcinającego
- odcinek leży częściowo wewnątrz, a częściowo na zewnątrz prostokąta obcinającego
- odcinek przecina prostokąt obcinający



Aby sprawdzić, który przypadek należy rozpatrywać, wystarczy zamiast wszystkich punktów odcinka sprawdzić jedynie jego dwa punkty – początkowy i końcowy. Jeżeli:

- oba punkty leżą wewnątrz prostokąta, otrzymujemy przypadek b) i możemy narysować cały odcinek
- jeden z punktów leży wewnątrz prostokąta, natomiast drugi na zewnątrz – przypadek c), należy znaleźć punkt przecięcia odcinka i prostokąta

- oba punkty leżą na zewnątrz prostokąta – przypadek a) lub d) – należy dodatkowo sprawdzić, czy odcinek nie przecina prostokąta

Sprawdzenie, czy odcinek przecina prostokąt polega na sprawdzeniu, czy odcinek przecina kolejno każdy z boków prostokąta (które też są odcinkami). Aby uprościć zadanie można obliczyć punkt przecięcia dwóch prostych – prostej na której leży odcinek i prostej na której leży dany bok prostokąta. Następnie wystarczy sprawdzić, czy obliczony punkt przecięcia spełnia nierówności (1.1) – jeżeli tak, odcinek przecina dany bok prostokąta.

Ogólne równanie prostej ma postać:

$$y = ax + b \quad (1.2)$$

proste w których znajdują się boki prostokąta można więc zdefiniować następująco (zgodnie z rysunkiem na początku instrukcji):

$$x = x_{\min}; \quad x = x_{\max}; \quad y = y_{\min}; \quad y = y_{\max}$$

Natomiast w celu wyznaczenia prostej (znalezienie współczynników a i b), na której leży rozpatrywany odcinek wystarczy rozwiązać układ równań, podstawiając w miejsca x i y wartości punktu początkowego i końcowego.

Następnie, gdy mamy już obliczone wzory obu prostych, w celu znalezienia punktu przecięcia należy rozwiązać poniższy układ równań:

$$\begin{cases} y = a_1 x + b_1 & \leftarrow \text{Równanie prostej na której leży bok prostokąta} \\ y = a_2 x + b_2 & \leftarrow \text{Równanie prostej na której leży odcinek} \end{cases} \quad (1.3)$$

1.4 Algorytm Cohena-Sutherlanda

Poniższy opis algorytmu został oparty o opis znajdujący się w książce: Michał Jankowski, „Elementy grafiki komputerowej”, WNT 2006

Ponieważ problem obcinania obrazu jest dosyć często spotykany, powstały algorytmy które upraszczają ten proces. Jednym z nich jest algorytm Cohena-Sutherlanda.

Podstawowym założeniem jest użycie dla punktów początkowego i końcowego odcinka czterobitowego kodu, określającego położenie względem prostokąta obcinającego. Kod dla punktu P ma następującą postać:

$$\text{kod}(P) = b_3 b_2 b_1 b_0, \text{ gdzie:} \quad (1.4)$$

$b_0 = 1$, gdy punkt leży na lewo od prostokąta,

$b_1 = 1$, gdy punkt leży na prawo od prostokąta,

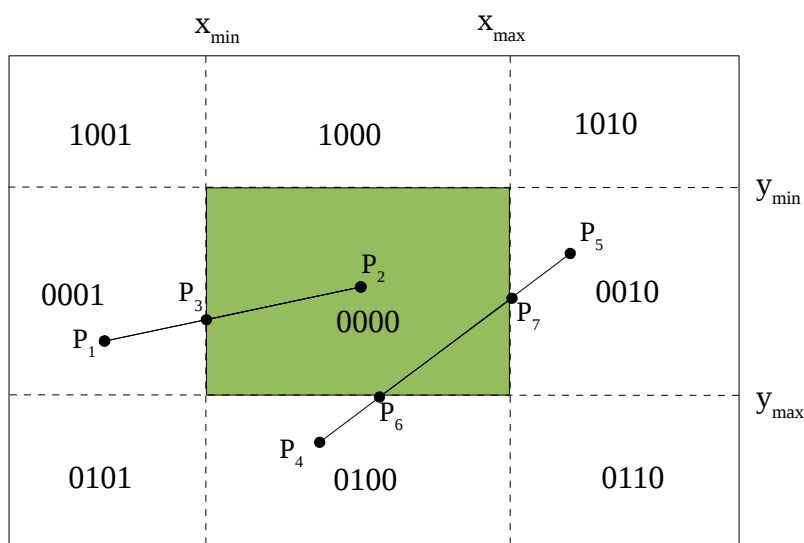
$b_2 = 1$, gdy punkt leży poniżej prostokąta,

$b_3 = 1$, gdy punkt leży powyżej prostokąta

Dla punktu leżącego wewnątrz prostokąta kod przyjmuje wartość 0000, natomiast pozostałe wartości gdy znajduje się poza nim. Dzięki tej właściwości można łatwo określić, czy cały odcinek znajduje się poza prostokątem. Wystarczy wykonać operację AND na bitach obu punktów odcinka:

$$\text{kod}(P_p) \text{ AND } \text{kod}(P_k) = abcd \quad (1.5)$$

Jeżeli otrzymany wynik (abcd) jest różny od 0000, wtedy odcinek leży na zewnątrz prostokąta. Jeżeli wynikiem jest 0000, oznacza to że odcinek leży w całości wewnątrz prostokąta (gdy kody obu punktów są równe 0000) lub najprawdopodobniej go przecina (jednak nie na pewno). W tym drugim przypadku należy znaleźć punkt przecięcia odcinka z prostą zawierającą bok prostokąta. Kolejnym krokiem jest zastąpienie końca odcinka o niezerowym kodzie obliczonym punktem, dzięki czemu odrzucamy część odcinka znajdującą się poza oknem. Kroki te należy powtarzać aż do momentu, w którym możemy stwierdzić że odcinek na pewno nie leży wewnątrz prostokąta lub na pewno w nim leży.



Przykład 1.

$\text{kod}(P_1) = 0001$, $\text{kod}(P_2) = 0000$ – jeden z kodów różny od 0000

$\text{kod}(P_1) \text{ AND } \text{kod}(P_2) = 0001 \text{ AND } 0000 = 0000$

Bit b_0 kodu punktu P_1 mówi nam, że punkt ten leży na lewo od prostokąta, zatem należy znaleźć

przecięcie z prostą zawierającą lewy bok prostokąta ($x=x_{\min}$). Punktem przecięcia jest punkt P_3 , którego kod to 0000 (zastępujemy zatem punktem P_3 punktem P_1). Kolejne sprawdzenie pokazuje, że odcinek P_3P_2 leży wewnątrz prostokąta obcinającego (natomiast P_1P_3 już nie):

$\text{kod}(P_3) \text{ AND } \text{kod}(P_2) = 0000 \text{ AND } 0000 = 0000$, oba kody równe 0000.

Przykład 2.

$\text{kod}(P_4) = 0100$, $\text{kod}(P_5) = 0010$ – oba kody różne od 0000

$\text{kod}(P_4) \text{ AND } \text{kod}(P_5) = 0100 \text{ AND } 0010 = 0000$

Bit b_2 kodu punktu P_4 mówi nam, że punkt ten leży poniżej prostokąta, zatem należy znaleźć przecięcie z prostą zawierającą dolny bok prostokąta ($y=y_{\max}$). Punktem przecięcia jest punkt P_6 , którego kod to 0000 (zastępujemy zatem punktem P_6 punktem P_4) i dokonujemy dalszego sprawdzenia:

$\text{kod}(P_6) \text{ AND } \text{kod}(P_5) = 0000 \text{ AND } 0010 = 0000$, jednak jeden z kodów nadal jest różny od 0000

Bit b_1 kodu punktu P_5 mówi nam, że punkt ten leży na prawo od prostokąta, zatem należy znaleźć przecięcie z prostą zawierającą prawy bok prostokąta ($x=x_{\max}$). Punktem przecięcia jest punkt P_7 , którego kod to 0000 (zastępujemy zatem punktem P_7 punktem P_5), sprawdzamy ponownie:

$\text{kod}(P_6) \text{ AND } \text{kod}(P_7) = 0000 \text{ AND } 0000 = 0000$, oba kody równe 0000

Otrzymaliśmy zatem obcięcie odcinka P_4P_5 , dzieląc go na jeden odcinek widoczny P_6P_7 i dwa niewidoczne: P_4P_6 i P_7P_5 .

Zapis krokowy algorytmu:

Założenia:

- końcami odcinka są punkty $P_1(x_1, y_1)$ oraz $P_2(x_2, y_2)$
- prostokąt obcinający jest ograniczony prostymi: $x=x_{\min}$, $x=x_{\max}$, $y=y_{\min}$, $y=y_{\max}$, zgodnie z poprzednimi oznaczeniami na rysunkach

Algorytm:

1. Wyznacz kody punktów P_1 i P_2 - $\text{kod}(P_1)$ i $\text{kod}(P_2)$
2. Sprawdź, czy $\text{kod}(P_1) = \text{kod}(P_2) = 0000$, jeśli tak zakończ algorytm, odcinek P_1P_2 leży wewnątrz prostokąta, jeśli nie przejdź do punktu 3

3. Sprawdź, czy $\text{kod}(P_1) \text{ AND } \text{kod}(P_2) \neq 0000$, jeśli tak zakończ algorytm, odcinek P_1P_2 leży w całości na zewnątrz prostokąta, jeśli nie przejdź do punktu 4
4. Wykonaj następujące kroki
 - a) sprawdź, czy $\text{kod}(P_1) = 0000$, jeśli tak zamień kody oraz punkty P_1 i P_2 między sobą
 - b) sprawdź bit b_0 punktu P_1 , jeśli $b_0 = 1$ to:
 - $y_1 = y_1 + (x_{\min} - x_1) * (y_2 - y_1) / (x_2 - x_1)$; $x_1 = x_{\min}$
 - przejdź do punktu 1
 - c) sprawdź bit b_1 punktu P_1 , jeśli $b_1 = 1$ to:
 - $y_1 = y_1 + (x_{\max} - x_1) * (y_2 - y_1) / (x_2 - x_1)$; $x_1 = x_{\max}$
 - przejdź do punktu 1
 - d) sprawdź bit b_2 punktu P_1 , jeśli $b_2 = 1$ to:
 - $x_1 = x_1 + (y_{\max} - y_1) * (x_2 - x_1) / (y_2 - y_1)$; $y_1 = y_{\max}$
 - przejdź do punktu 1
 - e) sprawdź bit b_3 punktu P_1 , jeśli $b_3 = 1$ to:
 - $x_1 = x_1 + (y_{\min} - y_1) * (x_2 - x_1) / (y_2 - y_1)$; $y_1 = y_{\min}$
 - przejdź do punktu 1

2 Zadanie

Napisać program, który implementuje algorytm obcinania odcinków. Program powinien rysować prostokąt obcinający na środku obszaru widocznego i wykonywać obcinanie dla dwóch dowolnych odcinków, 1 trójkąta złożonego z trzech odcinków i jednego prostokąta złożonego z 4 odcinków (wszystkie odcinki powinny być dobrane tak, aby figura mogła zmieścić się wewnątrz prostokąta). Każda z figur powinna być wyświetlana w losowym miejscu. Odcinki obcięte przez prostokąt powinny być zaznaczone na kolor zielony, natomiast odcinki odrzucone powinny być zaznaczone na kolor czerwony. W celu przyspieszenia działania programu można stosować funkcję Allegro do rysowania linii (prostokąt obcinający może być narysowany poprzez funkcję `rect()`).