

Laboratorium: Lista zadań 3
termin: 2021-06-13

1. (1p.) Przerób program
<https://mki1967.github.io/canvas-examples/cube-texture/cube-demo.html>
(Źródła możesz znaleźć w repozytorium:
<https://github.com/mki1967/canvas-examples>)
Zaprojektuj przy pomocy programu do rysowania (np. [GIMP](#) lub inny) obrazki i zastąp nimi tekstury w programie. (Pamiętaj o ustawieniu odpowiednich wymiarów obrazka.) Zademonstruj efekty stosowania różnych filtrów powiększających i pomniejszających (LINEAR / NEAREST). W funkcji `onKeyDown(e)` dodaj akcje umożliwiające użytkownikowi zmianę ustawień rodzajów filtrów pomniejszających i powiększających dla poszczególnych tekstur. Wykonaj wersję powyższego programu z wygenerowaniem mipmap. Postaraj się znaleźć sceny, na których widać różnice przy zastosowaniu mipmap i bez nich.
2. (1p.) Napisz program rysujący wykresy funkcji $R^2 \rightarrow R$, pozwalający je obracać i oglądać z różnych pozycji. (Do obracania i przesuwania możesz wykorzystać fragmenty kodu z przykładowych programów .) Wykresy mają dotyczyć kwadratowego fragmentu R^2 i być przybliżane wykresami kraty punktów o wymiarach co najmniej 500 x 500. Narysuj wykresy jako POINTS oraz jako przybliżenia powierzchni (np. TRIANGLES albo TRIANGLE_STRIP). Zademonstruj działanie programu dla różnych funkcji. Poczytaj o sposobach wyznaczania jasności na podstawie oświetlenia (np. na stronie <http://learnopengl.com/#!Lighting/Basic-Lighting>) i dorób oświetlenie powierzchni wykresu (przynajmniej składowe ambient i diffuse).
3. (0.5p.) Zrób tak aby kolory punktów wykresu bardziej oddalonych od obserwatora były bardziej wymieszane z kolorem tła. (Najlepiej to zrobić we fragmencie shader.)
4. (1p.) Wykorzystując `window.requestAnimationFrame` oraz stencil buffer, zaimplementuj animację przenikających się dwóch wykresów funkcji $R^2 \rightarrow R$, obracających się w przeciwnych kierunkach. Wykorzystaj stencil bufer do przepuszczania jednego obrazu w pikselach o współrzędnych (x,y) takich, że x+y jest parzyste a drugiego - na pozostałych pikselach.
Wskazówki:
 - Stencil buffer może być przygotowany jako procedura uboczna rysowania.
 - Do rysowania szachownicy pikseli można wykorzystać `gl_FragColor`.(Zobacz przykład:
<https://mki1967.github.io/canvas-examples/pixel-test/pixel-test.html>)
Wypróbuj przy innych partycjach pikseli (np. zależnie od parzystości jednej współrzędnej) i przy różnych jasnościach tła.
5. (1p.) Wykonaj poprzednie zadanie stosując blending zamiast stencil buffer. Dobierz odpowiednie sposoby mieszania obrazów.
6. (1p.) Wykorzystaj [ColorMask](#) do prezentacji wykresu w postaci [anaglifu](#) (obrazu stereoskopowego, który można obserwować przez czerwono-niebieskie okulary). Zrobić tak aby wykres można było obracać w sposób interaktywny.