Звіт з лабораторної роботи №5  
на тему «Quickhull»  
з дисципліни «Комп’ютерна графіка»  
студента 3-го курсу Факультету комп’ютерних наук та кібернетики   
групи ІПС-32  
Поліщук Єгора Даниловича

**Постановка задачі.**В просторі E2 задана множина точок S, яка містить N точок. Необхідно побудувати їх опуклу оболонку (повний опис границі ConvexHull(S)).  
  
**Розв’язання.**Розв’язком вважаемо упорядковану множину точок, оскільки за нею можна побудувати «відрізки» оболонки та відновити її у вигляді опуклого многокутника.  
  
Алгоритм розв’язання задачі є рекурсивним. Основна процедура – QuickHull(S, l, r), де S – множина точок, для яких шукається оболонка; l та r – крайні точки множини, які гарантовано належать оболонці та задають пряму розбиття.

***Алгоритм***:  
1. Задати пряму розбиття множини S двома точками l та r – умовно «ліва» та «права».  
l та r належать S.   
2. Якщо card(S) = 2, то S = {l, r} – повернути S.  
3. Визначити множини S1(точок «вище» lr – зліва вектора lr) та S2 (точок «нижче» lr – справа вектора lr).  
4. Визначити точку h з множини S1 найбільш віддалену від lr. Якщо знайдено більше одної такої точки, обирається та, для якої кут hlr більший. Точка h гарантовано належить шуканій оболонці.  
5. Визначити множини S11 (точок зліва від вектора lh) та S12 (точок зліва від вектора hr).  
6. Повернути зчеплений список quickHull(S11, l, h) + quickHull(S22, h, r).  
  
Задача опулої оболонки вирішується запуском функції quickHull(S, l0, r0) для S – множини точок, l0 – точки з множини S з мінімальною абсцисою, r0 – точки з координатами (l.x, l.y - eps), та видаленням точки r0 з отриманого списку;  
eps – деяке мале значення -> 0.  
Таким чином задається вектор l0r0, так що всі інші точки множини S лежать зліва від нього. Шукається деяка найбільш віддалена від прямої, заданої вектором, точка з цієї множини, тобто точка з максимальною абсцисою. Запускається та сама процедура для точок зліва від lh («над») та зліва від hr (тобто, фактично справа або ж знизу lh). Таким чином усі точки множини задіяні в алгоритмі.   
  
**Обосливості реалізації**  
1) «Сторони» прямих визначаються сторонами векторів.  
2) Найбільш віддалена точка для прямої AB визначається трикутником ABP з найбільшою площею, де P – довільна точка з множини. Це доведено для S=0.5 \* a\*ha, тут 0.5\*a – константа, тому S1>S2 ⬄ h1 > h2.  
3) Площа трикутника обчислюэться за формулою Герона.  
4) Кути між прямими вираховуються як кути між векторами з формули скалярного добутку.  
  
Мова реалізації: Java.  
Часова складність: у середньому O(NlogN), у гіршому випадку – O(N2).