French Azerbaijani University
Arash Habibi
Second year
2019-2020

# Data Structure and Algorithms 1

## Geometry exercises

The aim of these exercises is to calculate :

— the distance between two points in 2D ;

— the distance between a point and a line segment in 2D.

## 1 Vectors

Write a program with a header file `Vector.h`, a source file `Vector.c` and a main source file `main.c`. In the header file, define a vector as a structure characterized by two floating point numbers and, at least the two following functions :

— `Vector V_new(float x, float y);`
which returns a vector whose coordinates are respectively `x` and `y` ;

— `void V_show(Vector v, char *label);`
which prints on the terminal the coordinates of vector `v` possibly with an identifying label.

— `void V_diff(Vector v1, Vector v2);`
which returns a vector which equals the difference between `v1` and `v2`.

— `void V_mult(Vector v, float k);`
which returns a vector whose coordinates are those of `v` multiplied by `k`.

For example the following code :

```
Vector u = V_new(3,5);
Vector v = V_new(5,6);
Vector w = V_diff(v,u);
V_show(u, "u  ");
V_show(v, "v  ");
V_show(w, "v-u");
w = V_mult(w,-1);
V_show(w, "u-v");
```

should output something like :

```
u   : (3,5)
v   : (5,6)
v-u : (2,1)
u-v : (-2,-1)
```

## 2 The distance between two points

The *dot product* between two vectors $v_1(x_1, y_1)$ and $v_2(x_2, y_2)$ is a number defined as follows :

$$v_1 \cdot v_2 = x_1 y_1 + x_2 y_2 \tag{1}$$

1. Write function `float V_dotProduct(Vector v1, Vector v2);` which returns the value of the dot product of its arguments.
2. Write function `float V_magnitude(Vector v);` which returns the magnitude of vector `v`. The magnitude of vector $v$ equals $\sqrt{v \cdot v}$ (where $v \cdot v$ is the dot product of $v$ and itself).
3. Write function `float V_PtPtDistance(Vector A, Vector P);` which returns the distance between points $A$ and $P$. In order to do so, here are a few hints :
   — The distance between points $A$ and $P$ is the magnitude of the $AP$ vector ;
   — Vector $AP$ equals $P - A$ ;

# 3 The dot product

From a mathematical point of view, the dot product may also be defined in another manner. This does not mean that you have to change your dot product function. It only means that the same function can have other uses than simply calculate the length of a vector. If $\theta$ is the angle between vectors $v_1$ and $v_2$, then the dot product can also be defined in the following way :

$$v_1 \cdot v_2 = ||v_1||.||v_2||.cos\theta \tag{2}$$

where $||v_1||$ and $||v_2||$ represent the magnitudes (or lengths) of $v_1$ and $v_2$.

# 4 Distance between a point and a line

Equation (2) has several implications. In particular, if the magnitude of vector $v_2$ is 1 (in other words, if $v_2$ is a unit vector) then equation (2) becomes : $v_1 \cdot v_2 = ||v_1||cos\theta = l_1$ (see figure 1). Dot products are often used to calculate projections.

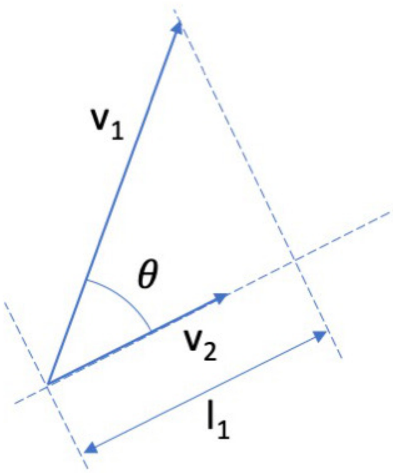


FIGURE 1: If $v_2$ is a unit vector, then the $v_1 \cdot v_2$ dot product is equal to the length $l_1$ of the projection of $v_1$ on the $v_2$ axis.

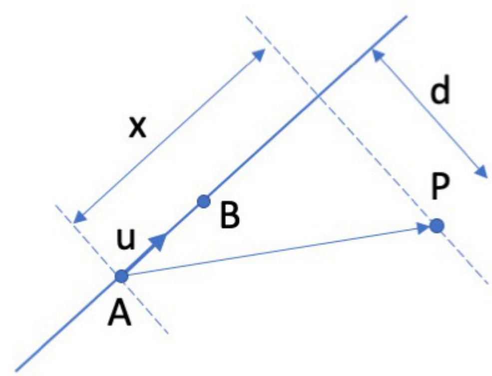


FIGURE 2: We know is the positions of $A$, $B$ and $P$ and we look for $d$, the distance between point $P$ and the $(AB)$ line.

1. Write function `Vector V_unit(Vector v);` which returns a unit vector parallel to `v`. If `v` is the null vector, the function must return an arbitrary unit vector (such as (1,0)). For example in figure 2, vector $u$ represents the unit vector corresponding to vector $AB$. It should be equal to `V_unit(V_diff(B,A))`.
2. Express $x$ using $A$, $P$ and $u$ (figure 2). This question and the two next ones do not require the computer. You can answer to them on a piece of paper or in your head if you like. You will simply need the answer for the last question, which does require to write a function.

3. Express $d$ (the distance between point $P$ and line $(AB)$) using $x$, $A$ and $P$.
4. Express $d$ using $A$, $B$ and $P$.
5. Write function `float V_PtLineDistance(Vector A, Vector B, Vector P);` which returns the distance between point $P$ and line $(AB)$.

## 4.1 Three regions

The distance between a point and a line segment can sometimes boil down to calculating the distance between a point and an infinite line (what we calculated in the previous section). But not always. Let us draw two lines perpendicular to the $(AB)$ line and passing through points $A$ (for the first line) and point $B$ (for the second line). These two lines are represented as dashed lines in figure 3. These two lines define three regions labeled I, II and III in the figure.

— If point $P$ belongs to region II, then the distance between $P$ and the $[AB]$ line segment is the same as the distance between $P$ and the $(AB)$ line.

— If point $P$ belongs to region I, then the distance between $P$ and the $[AB]$ line segment is the distance between $P$ and $A$.

— If point $P$ belongs to region III, then the distance between $P$ and the $[AB]$ line segment is the distance between $P$ and $B$.

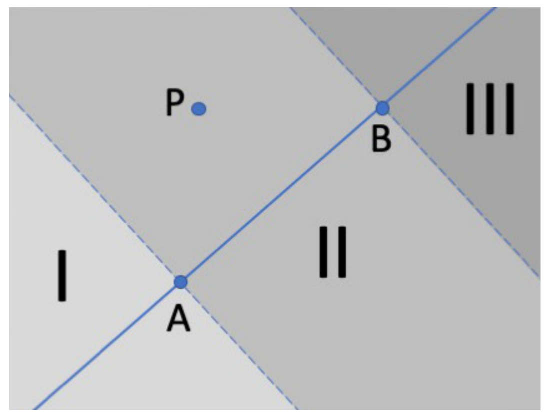


FIGURE 3: The distance between point $P$ and the $[AB]$ line segment depends on which region point $P$ belongs to.

This means that we have almost everything we need to calculate the distance between a point and a line segment. All we need is to determine the region to which belongs a given point $P$.

## 4.2 Acute angle

As illustrated in figure 4, equation (2) defining the dot product implies that :

— $v_1.v_2 > 0 \Leftrightarrow \theta < \pi/2$

— $v_1.v_2 < 0 \Leftrightarrow \theta > \pi/2$

— $v_1.v_2 = 0 \Leftrightarrow \theta = \pi/2$

This means that the dot product is an efficient indication of whether the angle between two vectors is acute or not :

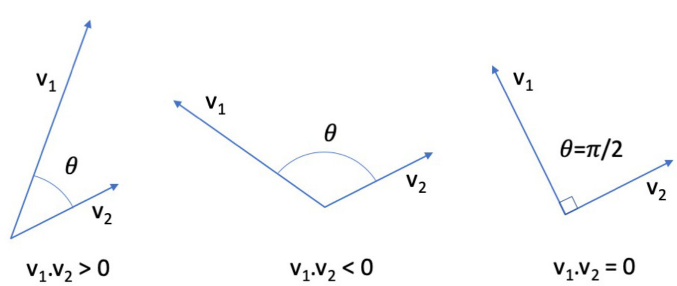


FIGURE 4: The dot product indicates rapidly whether the angle between two vectors is greater or smaller than $\pi/2$

Using the dot product, write function `bool V_isInRegionII(Vector P, Vector A, Vector B);` which returns `true` if $P$ is in region II and `false` if it belongs to region I or III.

# 5 Conclusion

Write and test function `float V_PtSegmentDistance(Vector P, Vector A, Vector B);` which returns the distance between point $P$ and the $[AB]$ line segment.