Laborator 2

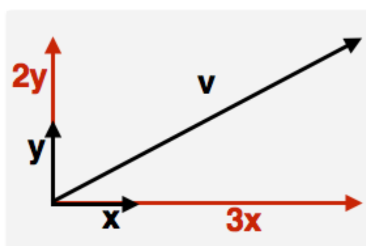
1 Obiective

Obiectivul acestui laborator este de a implementa un set de clase C++ pentru manipularea vectorilor.

2 Vectori

În domeniul graficii computerizate se utilizează vectori pentru a determina unghiul dintre muchii, orientarea suprafețelor, poziția relativă a unui punct pe o suprafață, calculul modelelor de iluminare și multe alte lucruri.

Vectorii se reprezintă printr-o listă de numere și grafic într-un sistem de coordonate cartezian. Reprezentăm vectorii ca săgeți și le numim folosind litere cu font **bold**. Din punct de vedere geometric, un vector este descris prin direcție și lungime. În 2D, un vector poate fi scris ca o combinație de doi vectori care nu sunt paraleli (cu o lungime diferită de 0). Un vector v este reprezentat de $\boldsymbol{v} = v_x\boldsymbol{x} + v_y\boldsymbol{y}$ unde v_x, v_y sunt coordonatele carteziene ale vectorului. Putem să scriem vectori pe orizontală și îi numim vectori rând sau îi putem scrie vertical și îi vom numi vectori coloană. Pentru exemplul anterior putem scrie: $\boldsymbol{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}$, or $\boldsymbol{v}^T = \begin{bmatrix} v_x & v_y \end{bmatrix}$.



În figura precedentă vectorul v poate fi definit ca o combinație a vectorilor de bază **x** și **y**.

$$\boldsymbol{v} = 3\boldsymbol{x} + 2\boldsymbol{y} = 3\begin{bmatrix} 1 \\ 0 \end{bmatrix} + 2\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

În domeniul graficii computerizate suntem interesați de vectori 2D, 3D și 4D și ne referim la elementele acestora folosind:

- x, y (în 2D)
- x, y, z (în 3D)
- x, y, z, w (în 4D)

Vectorii pot fi utilizați pentru a stoca deplasarea (decalajul dintre două puncte) sau locații (reprezentate ca deplasare dintr-o origine bine cunoscută). Rețineți însă că locațiile nu sunt vectori (nu putem adăuga "Cluj" la "București").

Diferența dintre două puncte este un vector ($\boldsymbol{v} = Q - P$), iar suma dintre un punct și un vector este un punct ($\boldsymbol{v} + P = Q$).

3 Operații

3.1 Modulul unui vector

Lungimea (modulul) este notată cu $\|\boldsymbol{v}\|$ și este egală cu rădăcina pătrată a sumei pătratului elementelor vectoriale. În cazul 2D, lungimea este $\|\boldsymbol{v}\| = \sqrt{v_x^2 + v_y^2}$.

Un **vector unitate** este un vector cu lungimea egală cu 1. **Vectorul zero** are lungimea egală cu 0 (în acest caz direcția este nedefinită).

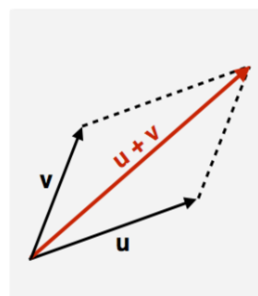
3.2 Normalizarea unui vector

Putem **normaliza** orice vector non-zero împărțind vectorul la lungimea sa. Noul vector este orientat în aceeași direcție cu vectorul original, dar lungimea lui este egală cu 1.

$$\boldsymbol{v} = \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}$$

3.3 Adunarea vectorilor

Putem aduna doi vectori însumând elementele corespunzătoare din vectorii inițiali. Dacă avem doi vectori 2D $\boldsymbol{u} = \begin{bmatrix} u_x \\ u_y \end{bmatrix}$ și $\boldsymbol{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}$ rezultatul operației de adunare este un vector egal cu:



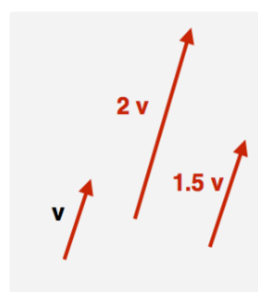
$$\boldsymbol{u} + \boldsymbol{v} = \begin{bmatrix} u_x + v_x \\ u_y + v_y \end{bmatrix}$$

Operația de adunare a vectorilor este comutativă:

$$\boldsymbol{u} + \boldsymbol{v} = \boldsymbol{v} + \boldsymbol{u}$$

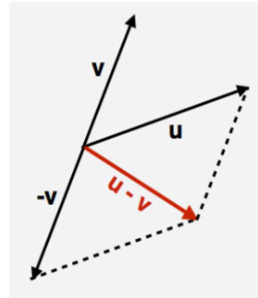
3.4 Scalarea vectorilor

Putem scala un vector cu o valoare scalară pentru a schimba lungimea vectorului. Pentru aceasta, înmulțim elementele vectorului cu valoarea scalară. Dacă scalăm un vector cu -1 schimbăm direcția vectorului.



$$\beta \boldsymbol{u} = \begin{bmatrix} \beta u_x \\ \beta u_y \end{bmatrix}$$

3.5 Scăderea vectorilor

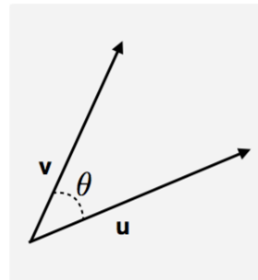


Scăderea a doi vectori este similară cu operația de adăugare, scalând cu -1 vectorul secund.

$$\boldsymbol{u} - \boldsymbol{v} = \boldsymbol{u} + (-\boldsymbol{v}) = \begin{bmatrix} u_x - v_x \\ u_y - v_y \end{bmatrix}$$

3.6 Înmulțirea vectorilor

3.6.1 Produsul scalar

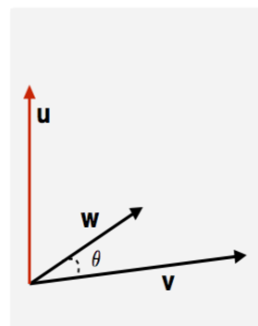


Produsul scalar (dot product) dintre doi vectori returnează o valoare scalară dependentă de lungimea vectorilor și de unghiul dintre ei.

$$\boldsymbol{u} \cdot \boldsymbol{v} = \|\boldsymbol{u}\|\|\boldsymbol{v}\| cos\theta$$

Dacă doi vectori **u** and **v** sunt reprezentați în coordonate Carteziene atunci $\boldsymbol{u} \cdot \boldsymbol{v} = \boldsymbol{u}_x \boldsymbol{v}_x + \boldsymbol{u}_y \boldsymbol{v}_y$. Similar, în 3D, produsul scalar este: $\boldsymbol{u} \cdot \boldsymbol{v} = \boldsymbol{u}_x \boldsymbol{v}_x + \boldsymbol{u}_y \boldsymbol{v}_y + \boldsymbol{u}_z \boldsymbol{v}_z$.

3.6.2 Produsul vectorial



În domeniul graficii computerizate folosim în principal acest produs numai pe vectori 3D, dar produsul poate fi generalizat. Rezultatul este un vector perpendicular pe cei doi vectori. Modului vectorului rezultat este $\|\boldsymbol{v} \times \boldsymbol{w}\| = \|\boldsymbol{v}\|\|\boldsymbol{w}\| sin\theta$. Direcția vectorului este perpendiculară pe planul format de cei doi vectori inițiali iar sensul se poate determina folosind regula mâinii drepte sau regula șurubului.

$$\boldsymbol{v} \times \boldsymbol{w} = \begin{bmatrix} \boldsymbol{v}_y \boldsymbol{w}_z - \boldsymbol{v}_z \boldsymbol{w}_y \\ \boldsymbol{v}_z \boldsymbol{w}_x - \boldsymbol{v}_x \boldsymbol{w}_z \\ \boldsymbol{v}_x \boldsymbol{w}_y - \boldsymbol{v}_y \boldsymbol{w}_x \end{bmatrix}$$

4 Temă

Descărcați codul sursă de pe pagina web a laboratorului. Trebuie să implementați metodele în interiorul fișierelor sursă (vec2.cpp, vec3.cpp și vec4.cpp). Fișierele antet (header) conțin definiția claselor și metodele care ar trebui implementate.