FONDAMENTI DI COMPUTER GRAPHICS M

LAB 01 – ADAPTIVE SUBDIVISION

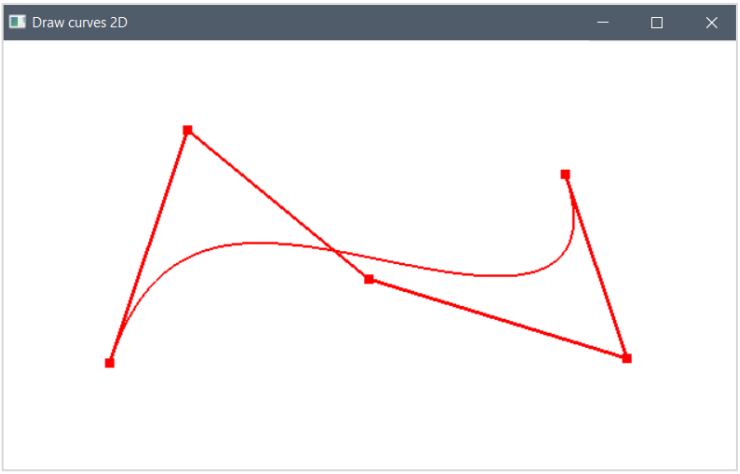
0. Test sulle funzionalità esistenti

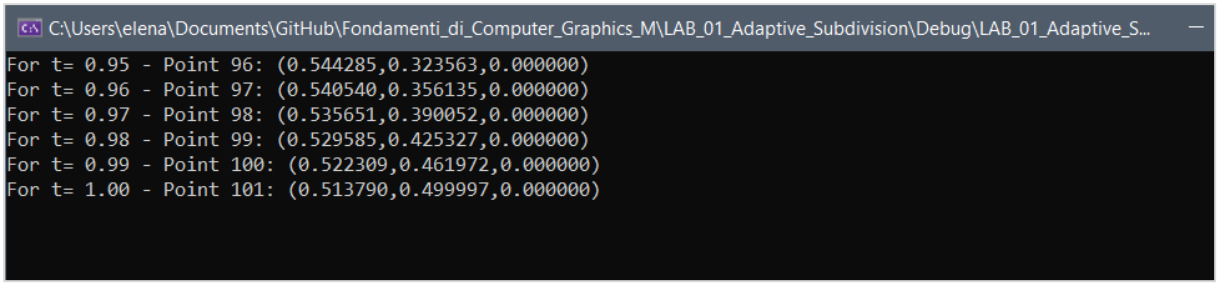
La prima parte del laboratorio (punti 1. e 2. della consegna) ha previsto l'utilizzo delle funzionalità già presenti nel codice di partenza fornito. In particolare, sono stati testati i controlli da keyboard ('f' ed 'l' rimuovono il primo e l'ultimo punto dalla lista di punti, rispettivamente) e da mouse (il left button aggiunge un punto alla lista, fino ad un massimo di 64 nuovi punti). Il programma, inoltre, disegna i punti e la poligonale che li connette.

Più approfonditamente, attraverso il codice, è stato possibile osservare l'utilizzo delle funzioni di callback OpenGL GLUT per la cattura di eventi di mouse e tastiera e la relativa associazione dei metodi da eseguire all'avvenire degli eventi.

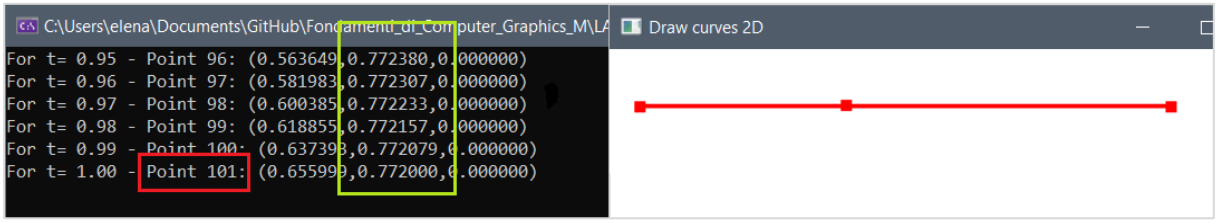
1. Utilizzo di de Casteljau

Il punto 3 della consegna ha richiesto di "*disegnare la curva di Bézier a partire dai punti di controllo inseriti, utilizzando l'algoritmo di de Casteljau*". Per questa prima richiesta, è stato sufficiente applicare la **discretizzazione uniforme** per cui si discretizza l'intervallo parametrico $t = [0,1]$, in N punti equidistanti, per ognuno di questi punti si applica de Casteljau ottenendo il punto sulla curva di Bézier e la curva mostrata a schermo è la poligonale che connette questi ultimi. Ovviamente, più alto è N più definita sarà la curva, ma più costoso sarà generarla.





Nell'esempio è stata utilizzata una variazione di t di *0.01*, che ha portato alla generazione di $N = 101$ punti tra cui anche il primo e l'ultimo punto della lista della poligonale iniziale (corrispondenti a $t = 0.0$ e $t = 1.0$). Questa soluzione permette di avere una precisione abbastanza alta nel rendering della curva, ma porta come svantaggio il fatto che il numero di punti sia prefissato, anche nel caso in cui la curva sia in realtà una semplice retta.



2. Suddivisione adattiva

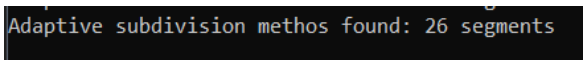
Il punto 4 della consegna ha richiesto di scegliere tra due alternative:

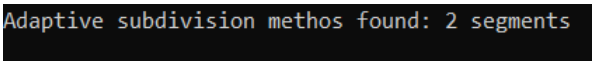
a) l'applicazione della suddivisione adattiva
b) gestione di curve di Bézier a tratti della relativa discontinuità.

È stato scelto il punto a), quindi il "disegno di una curva di Bézier mediante algoritmo ottimizzato basato sulla suddivisione adattiva". La suddivisione adattiva mira ad ottimizzare l'algoritmo precedente "adattando" il numero di punti (e quindi di segmenti) in base alle caratteristiche della curva. Si prevede che ad ogni iterazione, dati i punti di controllo che costituiscono la poligonale, si valuti la curva di Bézier per *t=0.5* e:

- se la distanza tra il punto calcolato ($P_{t=0.5}$) e il segmento che congiunge il primo e l'ultimo punto di controllo (P_0 e P_n) è inferiore a una determinata soglia, allora la curva descritta dai punti di controllo viene approssimata da tale segmento;
- altrimenti, dall'algoritmo di de Casteljau è possibile ricavare i punti della poligonale di controllo delle due sotto-curve generate. Quindi, si chiama ricorsivamente l'algoritmo di suddivisione adattiva per i nuovi set di punti.

Al termine dell'algoritmo, la curva sarà costituita da una serie di segmenti, ma a differenza del caso precedente il numero di segmenti dipenderà dalle caratteristiche della curva: là dove vi sia una curvatura maggiore il numero dei segmenti sarà maggiore (al contrario, in un caso estremo come quello di curvatura assente i segmenti saranno pochi). Questo consente di avere una buona resa grafica (che dipende dalla soglia impostata) e un numero molto minore di segmenti.





3. Modifica dei punti

Per poter modificare i punti tramite trascinamento, per prima cosa è necessario poter individuare quale punto di controllo è soggetto al trascinamento:

- ad ogni movimento del mouse (passivo, quindi senza la pressione di eventuali pulsanti), tramite la funzione di callback definita per `glutPassiveMotionFunc` di OpenGL, viene monitorata la posizione del cursore. Nel caso in cui il cursore sia vicino a un punto (distanza minore di una determinata soglia), quest'ultimo viene etichettato come punto selezionato;
- Alla pressione continua del *left button* del mouse e ad un movimento del mouse associato, invece, viene attivata la funzione di callback definita per `glutMotionFunc`, con la quale andiamo a modificare le coordinate del punto selezionato.