

LearnGraphs

**Budișteanu Ionuț Alexandru – CN „Mircea cel Bătrân” Vâlcea, elev, realizator,
ionutbudisteanu@yahoo.com**

**Mlisan Mirela – CN “Mircea cel Bătrân” Vâlcea, profesor îndrumător,
mirela_mlisan@yahoo.com**

Abstract

LearnGraphs este un soft educațional ce vine în sprijinul elevilor dar în același timp și al profesorilor în munca lor cu elevii. Aplicația se dorește a fi un soft foarte ușor de utilizat. Se dorește ca prin folosirea acestui soft procesul de studiere, verificare și însușire a noțiunilor referitoare la teoria grafurilor, să fie cât mai atrăgător și eficient, deoarece se cunoaște că teoria grafurilor nu este atât de ușor de predat și însușit, fiind o parte a programării bogată în noțiuni teoretice. Softul este împărțit în 4 aplicații „LearnGraphs – Grafuri Neorientate”, „LearnGraphs – Grafuri Orientate”, „LearnGraphs – Arbori binari” și „LearnGraphs – Sistem de evaluare”.

1. Introducere

LearnGraphs este un software educațional, ce dorește ca orele în care se predă capitolul grafuri să fie mult mai lejere și mai atrăgătoare pentru elevi. Este deja o certitudine faptul că elevii înțeleg mai ușor o lecție nouă dacă au ocazia să interacționeze cu un soft asemănător AELului.

De la această premisă a pornit ideea mea. Avantajul este că fiecare elev poate vizualiza, modifica, și chiar studia comportamentul unui algoritm pe un graf creat de el însuși, poate interacționa cu grafuri deja salvate sau importate, fapt care nu este la îndemâna unui manual.

Prima versiune a acestui software a luat naștere în luna ianuarie 2009. De atunci aplicația a trecut pe la mai multe concursuri, iar eu fiind receptiv la sugestiile propuse, programul a suferit multiple îmbunătățiri. Cel mai important pentru mine este faptul că aplicația a fost testată de mulți profesori, studenți, elevi, chiar și de colegii mei. Feedback-ul primit a contribuit la îmbunătățirea conținutului softwareului.

2. Caracteristici generale

Pentru că acest soft se adresează elevilor și profesorilor, a trebuit să îmbinăm câteva elemente opuse. A fost o adevărată aventură să creăm un soft simplu de folosit, intuitiv dar în același timp să funcționeze atât pentru Grafuri Orientate, Grafuri Neorientate cât și pentru Arbori binari. Eu mi-am propus ca utilizatorii noștri să parcurgă toată materia cu cât mai mulți algoritmi studiați, în cel mai plăcut și rapid mod posibil, astfel încât navigarea în mediul virtual oferit să fie o plăcere și nu o bătaie de cap.

Soft-ul are la bază o clasă TDrawClass, care desenează pe canvas obiectele(din vectorul alocat dinamic VertexContainer) și desenează muchii/arcele(din vectorul alocat dinamic EdgeContainer) .

Acum câteva cuvinte despre aplicația în sine. Profesorul nu trebuie decât să aducă un video – proiector sau să distribuie gratuit soft-ul elevilor și să-l predea în cel mai plăcut mod, folosind lecțiile oferite. Întrucât aplicația are și o ramură autodidactă, elevul va învăța mult mai ușor prin descoperire și mai ales prin interactivitatea cu lecția. Pe lângă lecții și partea de simulare, aplicația mai oferă un sistem de verificare a cunoștințelor pe mai multe niveluri de complexitate, iar după rezolvarea testului aplicația îi oferă utilizatorului posibilitatea de a vedea unde a greșit și a se corecta. Cel mai important lucru pentru elev este că primește la fiecare întrebare un feedback, aplicația oferind chiar soluția.

3. Despre editorul software-ului

După cum am menționat și mai sus, proiectul este creat din mai multe părți. Ca să le deschidem mai ușor, vom porni executabilul “Loader.exe” și vom da click pe unul dintre butoane(a se vedea in Figura 1)

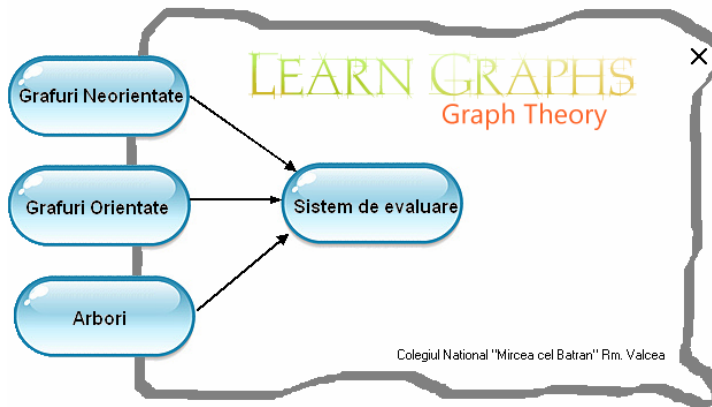


Figura 1. Interfața executabilului “Loader.exe”

Cea mai importantă parte a proiectul este editorul de grafuri. El permite crearea grafurilor, cât mai ușor, și sugestiv. Crearea se face cu drag&drop. Deci vom da click pe „Nod”(din meniul „Editor”, partea dreaptă) și vom duce cursorul pe suprafața de desenare și la eventul onMouseDown se inserează în vectorul alocat dinamic, VertexContainer nodul, la poziția curentă a cursorului.

Opțiuni globale la noduri	Opțiuni globale pt. arce	Opțiuni globale pt. etichete
Nume	Nume	Nume
Tip Nod	Nodul sursă	Poziție X
Poziție X	Nodul destinație	Poziție Y
Poziție Y	Grosime linie	Culoare
Culoare fundal	Culoarea liniei	Mărime
Culoare linie	Pondere asociată arcului	Text
Comentariu(o eticheta)		
Rază		

Adăugarea unei muchii/arce se face aproape similar. Vom da click pe butonul “Adaugă muchie” (din meniul „Editor”, partea dreaptă), vom selecta Nodul sursă, și după aceea trebuie să selectăm nodul destinație. Muchia se adaugă automat, în vectorul EdgeContainer.

Programul permite, adaugarea etichetelor, în anumite poziții.

Toate obiectelor au anumite proprietăți, setate manual din „ObjectInspector”(in partea stângă) în cazul în care nu suntem satisfăcuți de cele generate automat.

4. “LearnGraphs – Grafuri Neorientate”

Algoritmi implementati pentru studiul lor.

1. Matricea de adiacență
2. Matricea ponderilor
3. Lista de incidentă

4. Lista arce - noduri
5. Matricea drumurilor(Roy – Warshall)
6. Matricea vârfuri arce
7. Parcurgere in lățime(vizualizare + plus listă)
8. Componente conexe(vizualizare + plus listă)
9. Parcurgere in adâncime(vizualizare + plus listă)
10. Determinare lanț(vizualizare + plus listă)
11. Determinare ciclu(vizualizare + plus listă)
12. Graf bipartit(vizualizare + plus listă)
13. Arbore parțial de cost minim(vizualizare + plus listă, algoritmul lui Prim)
14. Drumul de cost minim(vizualizare + plus listă , algoritmul lui Dijkstra)

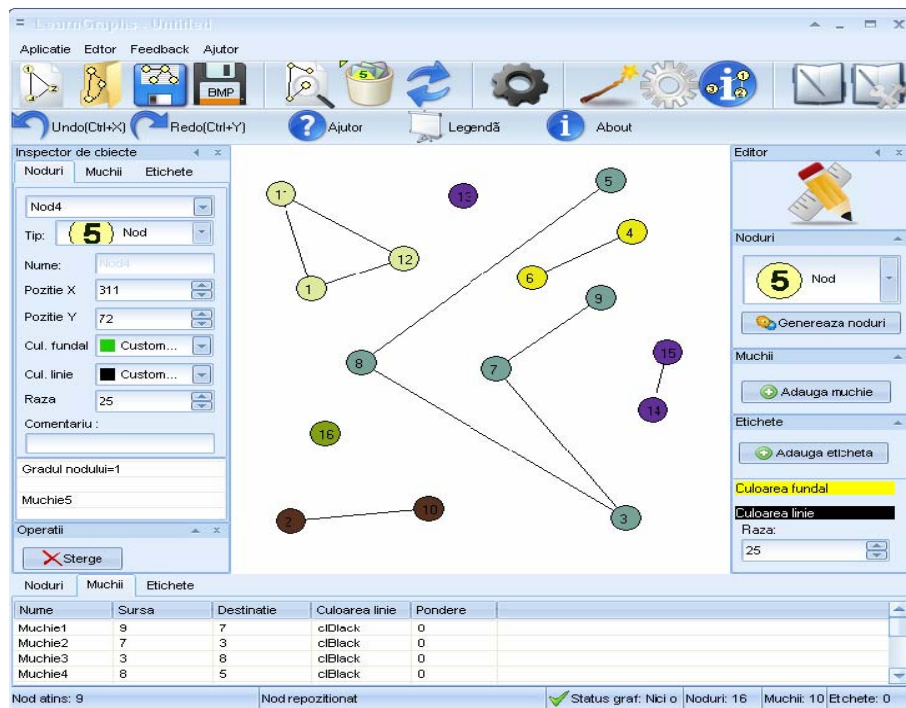


Figura 2. Interfața editorului („LearnGraphs – Grafuri neorientate”)

5. “LearnGraphs – Grafuri Orientate”

Algoritmi implementati pentru studiul lor.

1. Matricea de adiacență
2. Matricea ponderilor
3. Lista de incidentă
4. Lista arce - noduri
5. Matricea drumurilor(Roy – Warshall)
6. Matricea vârfuri arce
7. Parcurgere in lățime(vizualizare + plus listă)
8. Componente conexe(vizualizare + plus listă)
9. Parcurgere in adâncime(vizualizare + plus listă)

10. Determinare lanț(vizualizare + plus listă)
11. Determinare ciclu(vizualizare + plus listă)
12. Graf bipartit(vizualizare + plus listă)
13. Drumul de cost minim(vizualizare + plus listă , algoritmul lui Dijkstra)
14. Sortare Topologică (vizualizare + plus listă)
15. Determinare ciclu eulerian(vizualizare + plus listă)
16. Determinare ciclu hamiltonian(vizualizare + plus listă)

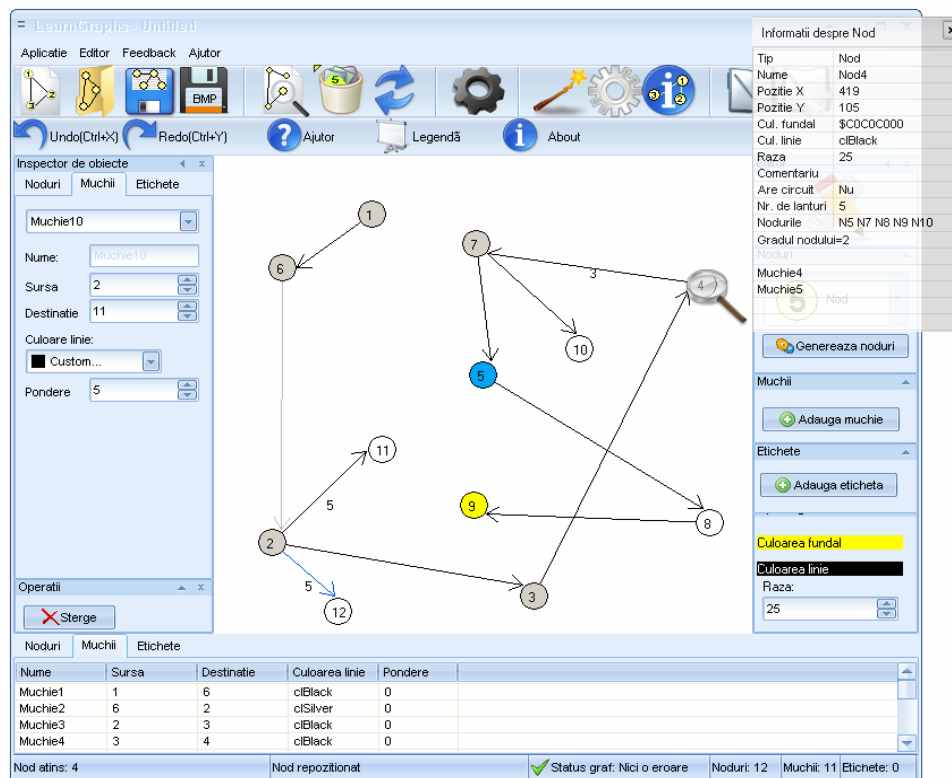


Figura 3. Interfața editorului

(„LearnGraphs – Grafuri Orientate”, parcurgere în adâncime din Nodul 1, nodul curent vizitat, fiind nodul 5)

6. “LearnGraphs – Arbori binari”

Algoritmi implementați pentru studiul lor.

1. Parcurgere în preordine (vizualizare + plus listă)
2. Parcurgere în inordine (vizualizare + plus listă)
3. Parcurgere în postordine (vizualizare + plus listă)
4. Parcurgerea pe niveluri (vizualizare + plus listă)
5. HeapSort
6. MaxHeap (creări, inserari, ștergeri etc)
7. MinHeap (creări, inserari, ștergeri, etc)

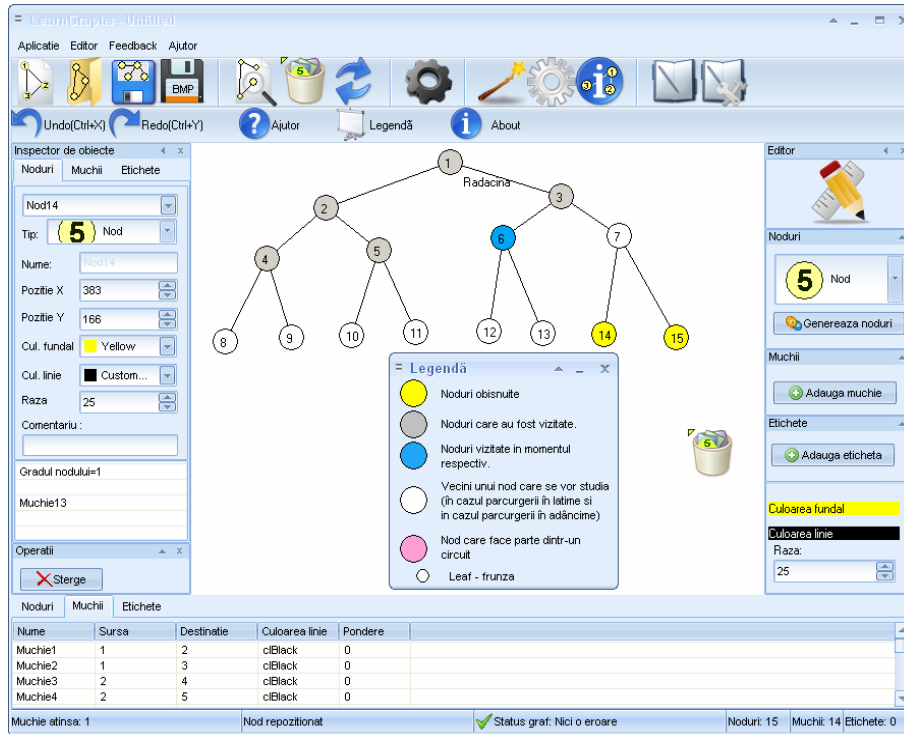
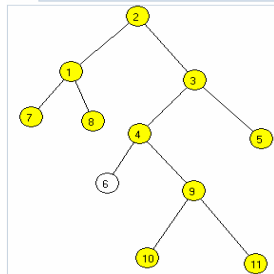
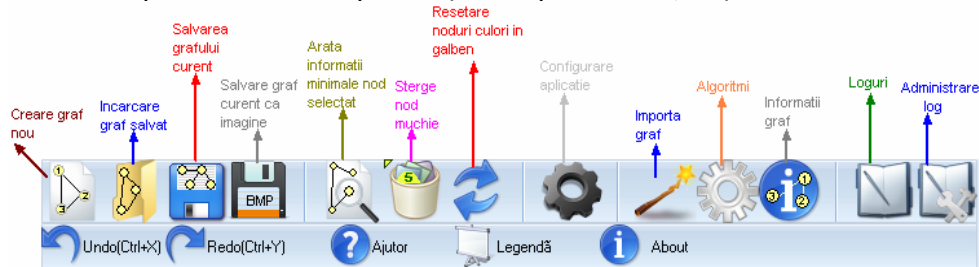


Figura 3. Interfața editorului („LearnGraphs – Arbori binari”, parcurgere pe nivele)

7. Manual de folosire

ToolBar – reprezintă bara cu care putem acționa asupra editorului, conține:



Suprafața editorului – reprezintă partea în care putem să cream nodurile, muchile, etichetele. El se găsește în centrul aplicației.

Figura 3. Un graf desenat în suprafața de lucru

Meniul cu care putem adăuga obiecte(figura 4) – puteam să creăm noduri, muchi, etichete. Se găsește în partea dreaptă a aplicației.

Inspectorul de obiecte(figura5)– puteam modifica, proprietățile nodurilor, muchiilor, și a etichetelor. Se găsește în partea stângă a aplicației.

Aceasta este una dintre cele mai importante facilități, a softului, astfel după ce am adăugat nodurile, și muchile pe parcurs, putem modifica, proprietățile obiectelor. Se pot modifica toate proprietățile obiectelor menționate mai sus.



Figura 4. Meniul de adaugă obiecte

Figura 5. Inspectorul de obiecte

Starea grafului – puteam vizualiza starea grafului, dacă avem greșeli în graful creat, Exemplu: avem 5 Noduri, și o muchie, destinație îl are pe nodul 7, normal este o greșeală. (StatusBar-ul se găsește în partea jos a aplicației).



Figura 6. StatusBar-ul

7. LearnGraphs – Sistem de evaluare

Proiectul, are încorporat, o parte educațională, în care sunt prezente definiții, și teste referitoare la materie. Am dorit, să fie foarte puțină, materie, deoarece nu aceasta este scopul, ci învățarea,

exersand. În partea aceasta, sunt doar $\frac{3}{4}$ din toți algoritmi scriși, deci cei mai fundamentali. După ce am studiat lecțiile, și am făcut la fiecare lecție testele specifice, avem un „așa numit” Test Recapitativ, prin care se trece prin toată materia, la capitolul respectiv.

Țin să menționez că doar definițiile, din această parte, sunt luate din manualele din bibliografie. Testele și soft-ul sunt creat exclusiv de mine.

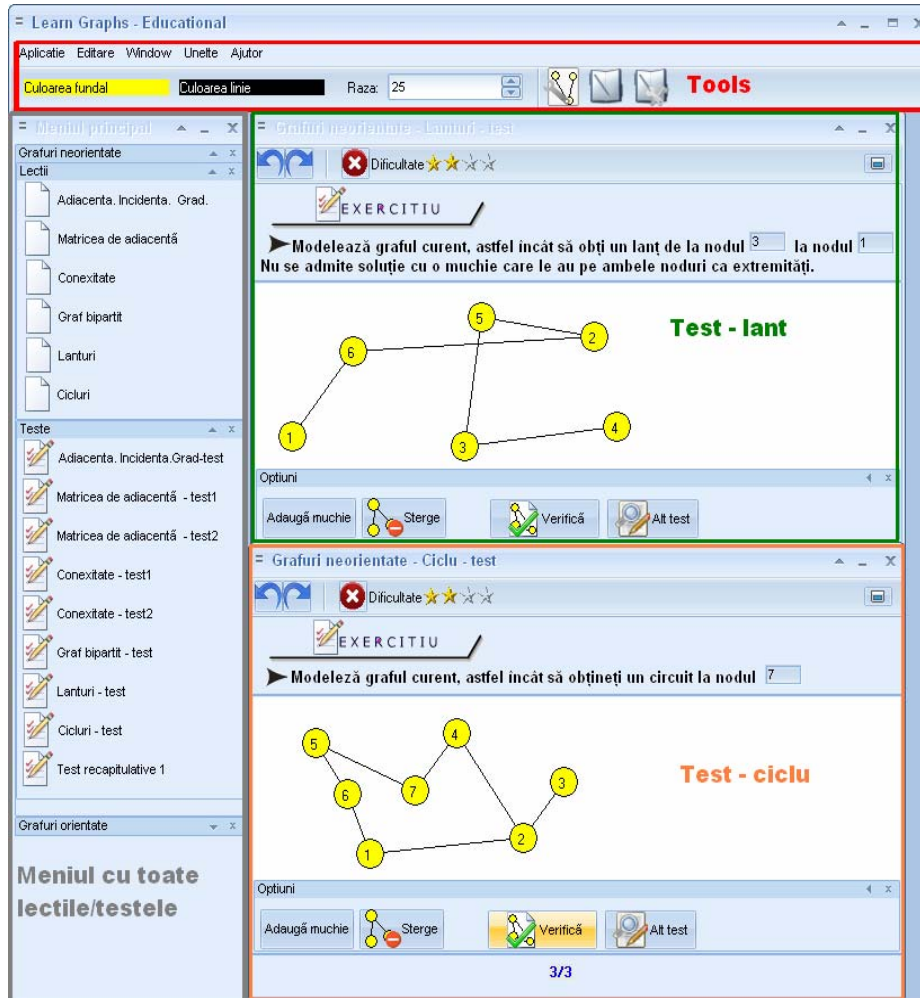


Figura 7. Sistem de evaluare

9. Inovații

Până acum probabil unele dintre opțiunile și caracteristicile pe care le-am enunțat mai sus le-ați mai întâlnit și în alte softuri educaționale care s-au ocupat de același subiect. Am rezervat acest capitol pentru elementele speciale din acest proiect.

- Modelarea proprietăților nodurilor
- Modelarea proprietăților muchiilor

- Modelarea proprietăților etichetelor
- Importul și exportul în mai multe formate
- Salvarea și încărcarea grafului.
- Salvarea în format bitmap (*.bmp)
- Help dinamic (MS Agent 2.0), asigură asistență pe parcursul rulării.
- Crearea până la 1000 de noduri, adăugarea a până la 1000*999 muchii
- Profesori, pot utiliza afisarea grafulor, și algoritmi implementați, doar printr-o simplă instanță, într-o posibilă aplicație viitoare.

10. Concluzii

Pentru comparație, prezint un link cu un program comercial asemănător (yED), cu unul dintre cele mai interesante după piață[dar doar editor] http://www.yworks.com/en/products_yed_about.html) Care totuși este un software, de modelare a grafulor, și nu de simulare a diverșilor algoritmi informatici, utili în studiul predării materiei grafuri la nivel de învățământ preuniversitar. Tehnologii folosite:

- Borland Delphi 6
- BusinessSkin 6.50(o librărie, care face skin-ul, nu este scrisă de mine)
- Microsoft Agent 2.0(importat ca ActiveX)
- Restul, sunt componente standard.

Concepte care stau la baza, realizari programului

1. Programare concurenta, utilizez intensiv threaduri, pentru algoritmi.
2. Tehnici, de gestionarea erorilor.
3. Programe top-down., algoritmica:
 - Recursivitate
 - Divide et impera
 - Programare dinamică
 - Arbori binari
 - Liste înlănțuite și dublu înlănțuite(alocare dinamica)
 - Geometrie computațională(geo. analitică) [pentru desenarea grafului]
4. Double Buffer

BIBLIOGRAFIE

- [1] Tudor Sorin, „Manual informatica, clasa a XI-a”
- [2] Cristian Udrea, „Manual informatica, clasa a XI-a”
- [3] E. Cerchez, M. Serban „Programare in limbajul C++ pentru Liceu”, V. 3, e. Polirom
- [4] Thomas H. Cormen, Charles Leiserson, Ron Rivest „Introduction to Algorithms”
- [5] Ioan Tomescu – „Combinatorica și grafuri”
- [6] C. Croitoru – „Algoritmica grafulor”