

Z DOŚWIADCZENIA OPRACOWANIA OPROGRAMOWANIA EDUKACYJNEGO

**Michaił Lvov,
Oleksandr Spivakovski,
Władzisław Kruglik,
Natalia Kushnir,
Anton Grabowski**

**Chersoński Państwowy Uniwersytet
kushnir@ksu.ks.ua
www.ksu.edu.ua**

Wstęp

Reformowanie systemu oświaty na Ukrainie, ogłoszono zadaniem priorytetowym Państwa. Przewiduje się zmianę jakościową technologii kształcenia na podstawie technologii informacyjnej. Technologie informacyjno-komunikacyjne z jednej strony są potężnym narzędziem do otrzymania różnej informacji przez dziecko, z drugiej – efektywnym środkiem dla podniesienia zainteresowania do nauki, i również motywacji, pogłębienia, naukowości i in.

W celu badania efektywności wykorzystania TIK w zarządzaniu instytucjami oświaty, w procesie nauczania, opracowania i wdrażania oprogramowania edukacyjnego i systemu kształcenia dystansowego w 2004 roku na Chersońskim Państwowym Uniwersytecie był utworzony Instytut Naukowo-Badawczy Technologii Informacyjnej (INB TI), który połączył wszystkie istniejące laboratoria.

Projekty edukacyjne takie jak ZPM “Wideointerpreter algorytmów wyszukiwania i sortowania”, Zestaw Programowo-Metodyczny TerM, Środowisko programowe „Układy równań liniowych” przeszły certyfikację na Ukrainie i polecane przez Ministerstwo Oświaty i Nauki do wykorzystania w procesie nauczania.

Zestaw programowo-metodyczny “Wideointerpreter algorytmów wyszukiwania i sortowania”

ZPM “Wideointerpreter algorytmów wyszukiwania i sortowania” jest przeznaczony do wykorzystania przy nauczaniu podstaw algorytmizacji oraz programowania w szkole. Pomoc programowa została zrealizowana w językach ukraińskim, rosyjskim i angielskim.

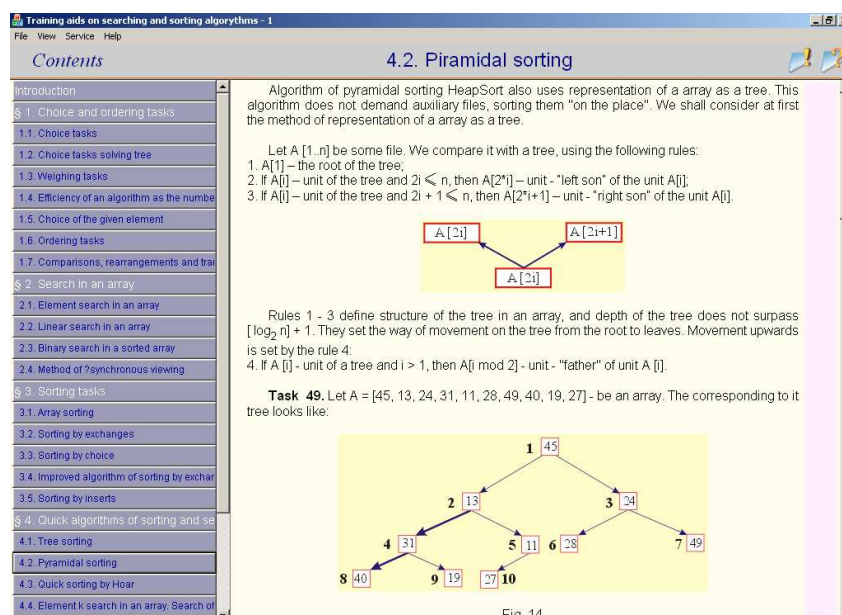
Aby zacząć pracę, użytkownik powinien się zalogować. W tym celu wybiera on klasę oraz swoje nazwisko w ZPM. Po zalogowaniu się otrzymuje on dostęp do następujących modułów:

- *Środowisko programowania* z listą algorytmów ZPM oraz kolekcją algorytmów użytkownika.
- *Środowisko demonstracji* z listami demonstracji algorytmów ZPM i użytkownika.
- *Podręcznik elektroniczny* z algorytmizacji „Algorytmy wyszukiwania i sortowania”.
- *Informator elektroniczny* z języka programowania Pascal.



Podręcznik elektroniczny z algorytmizacji jest nowoczesnym podręcznikiem elektronicznym, który zawiera fragmenty tekstowe, ilustracje graficzne tworzenia algorytmów, inne obiekty graficzne, teksty algorytmów. Podręcznik składa się z czterech rozdziałów:

1. Zadania wyboru i porządkowania.
2. Poszukiwanie w masie.
3. Zadania sortowania.
4. Szybkie algorytmy poszukiwania i sortowania.

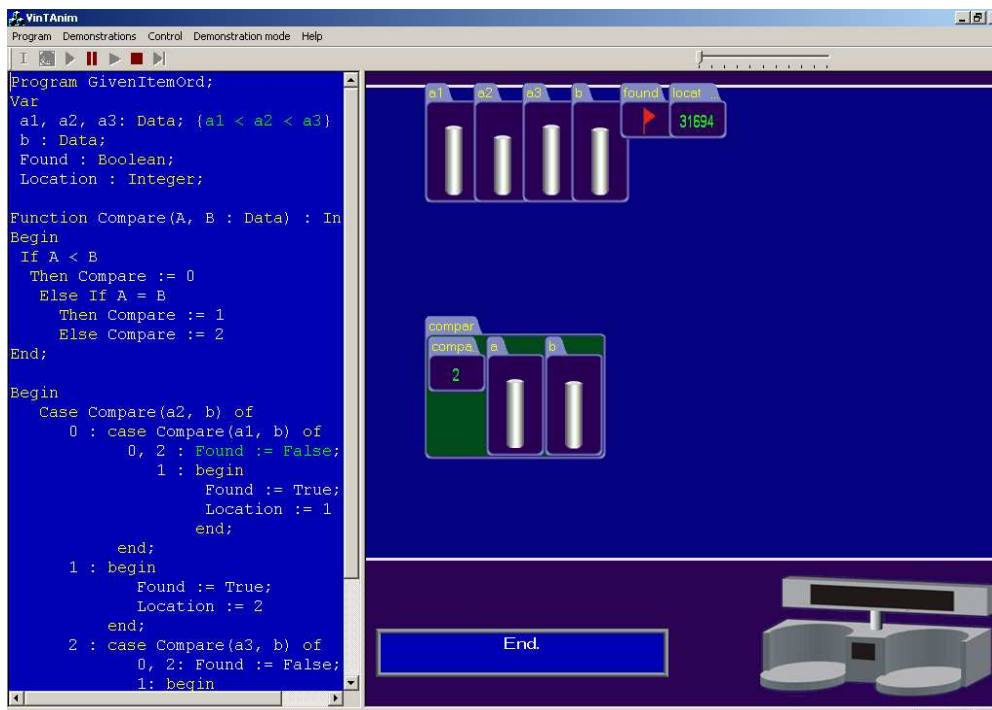


Na marginesach podręcznika użytkownik może robić własne notatki. Zachowują się one w kolekcji notatek danego użytkownika.

Czytelnik podręcznika może śledzić tworzenie algorytmu, który go zainteresował, przy pomocy polecenia *Demo*.

Otwiera się wtedy środowisko demonstracji z pełnym tekstem algorytmu w lewej części. Aby rozpocząć pracę w środowisku demonstracji, należy wykonać polecenie *Rozpocząć*. W efekcie na ekranie pojawią się obiekty, w których zostaną pokazane nazwy oraz znaczenia zmiennych programu – obiekty-zmienne, a także obiekty demonstrujące procedury i funkcje programu – obiekty-podprogramy. Istnieje na życzenie możliwość rozmieszczenia tych obiektów na ekranie. Dalej wykonamy generację danych wejściowych. W specjalnym oknie dokonujemy korekty generacji – wybieramy zmienne i do każdej z nich wyznaczamy

formę generacji. Naciskamy przycisk Apply. Trwa proces generacji – animowany generator wypełnia wartościami nasze zmienne.



Program jest gotowy do startu na wykonanie, więc zrobmy to!

Co widzimy? W lewej części podświetla się tekst algorytmu, w prawej zaś trwa jego animowane wykonanie. Użytkownik może przeprowadzić korektę demonstracji:

- zmienić prędkość demonstracji
- zmienić tryb demonstracji algorytmu (krok po kroku lub ciągle) oraz wykonawcy (pokazywać lub nie przesyłanie albo porównywanie).

Użytkownik może zrobić przerwę na dowolnym etapie tworzenia algorytmu, a po niej kontynuować pracę. Albo w ogóle zatrzymać wykonanie algorytmu, lecz nie będziemy tego robić. Lepiej spojrzmy na wynik...

Wykorzystanie dynamicznych obrazów operacji przyswajania, porównywania, przekazywania parametrów do procedur i funkcji, rekursywnych wywołań procedur i funkcji, procesu generacji danych wejściowych czyni środowisko demonstracji wyjątkowo pożytecznym środkiem nauczania podstaw algorytmizacji.

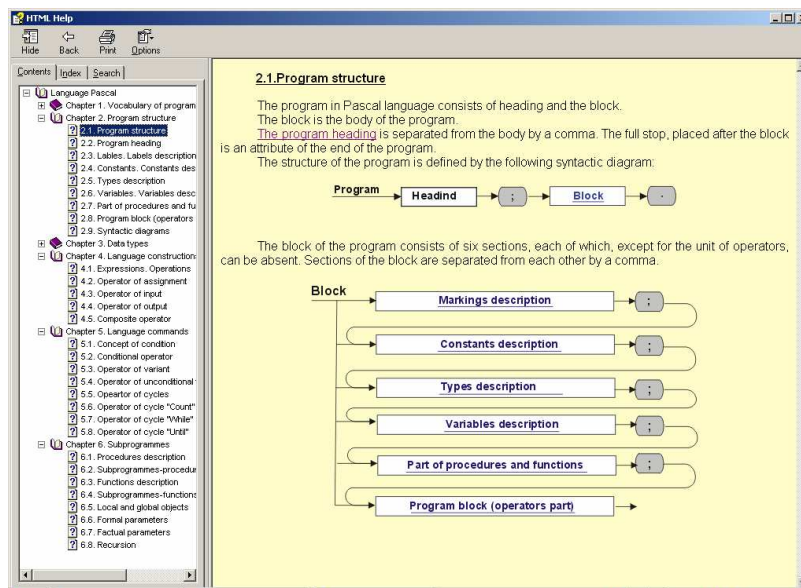
Zamknijmy program.

Przejdźmy do środowiska programowania

Użytkownik ma dostęp do listy układu z 25 algorytmów, których szczegółowy opis mieści się w podręczniku elektronicznym ZPM. Istnieje możliwość otwarcia tekstu algorytmu albo włączenia algorytmu na wykonanie w środowisku demonstracji.

Ale najważniejsze jest to, że istnieje możliwość tworzenia własnych algorytmów dla środowiska demonstracji.

Napiszmy program do znalezienia większej z dwóch danych liczb. Tworzymy nowy dokument. Proszę spojrzeć: po prawej stronie są szablony podstawowych struktur językowych do napisania programu. Jest potrzebny szablon do programu. Wykonaliśmy go. Opiszmy zmienne programu. W razie zapomnienia sposobu tworzenia sięgnijmy po pomoc kontekstową. Otworzy się Informator elektroniczny z języka programowania Pascal. Jest osobnym modulem, który zawiera wiadomości o języku programowania Pascal.



Przypomnieliśmy sobie! Opiszmy te zmienne z typem specjalnym DATA. To jest własny typ danych ZPM Wideointerpreter. Jest on wykorzystywany do demonstracji sporządzania algorytmów.

Teraz napiszemy kod podstawowy programu. Skorzystamy z odpowiedniego szablonu... Już, napisanie programu zakończono.

Zachowamy algorytm ten we własnej liście – liście użytkownika. Dalej uruchomimy demonstrację i włączymy ją na wykonanie.

Sformujemy dane i włączymy program na wykonanie. Większy element mieści się w zmiennej Max. To wszystko.

Obecnie jest realizowana wizualizacja wykonania algorytmów w innych dziedzinach przedmiotowych.

TerM

Podstawowym przeznaczeniem Zestawu Programowo-Metodycznego TerM jest wykorzystanie komputera na zajęciach praktycznych oraz w pracach laboratoryjnych z matematyki, tzn. aktywnej działalności matematycznej użytkownika. Obecnie ZPM jest praktycznym uzupełnieniem kursu „Algebra 7”.

ZPM TerM VII zrealizowano w trzech językach – ukraińskim, rosyjskim i angielskim. Na początku pracy z ZPM TerM VII użytkownik musi zalogować się. Wybiera on klasę, swoje nazwisko oraz zeszyt.

Następnie użytkownik otrzymuje dostęp do następujących modułów układu:

- *Pomoc naukowa (Textbook)*
- *Informator (Handbook)*
- *Zbiór zadań (Taskbook)*
- *Zeszyt (Copybook)*
- *Środowisko rozwiązywania (Ambience of decision)*
- *Wykresy (Graphs)*
- *„Rozwiązywacz” (Solver)*

Pomoc naukowa jest nowoczesnym podręcznikiem elektronicznym, w którym przedstawiono materiał teoretyczny. Omówieniu tego materiału towarzyszy moduł programowy *Ćwiczenia (Exercises)*.

Ćwiczenia są przeznaczone do samodzielnego sprawdzania wiedzy uczniów oraz nabycia przez nich umiejętności i praktycznych nawyków obliczania i wykorzystywania transformacji algebraicznych w trakcie rozwiązywania podstawowych zadań na dany temat.

Każde ćwiczenie zawiera trzy przykłady tego samego typu.

System sprawdza poprawność odpowiedzi i pokazuje wynik na sygnalizatorze świetlnym.

Kolor czerwony oznacza, że przykład został rozwiązany nieprawidłowo, żółty – poprawnie, ale nie jest uproszczony do końca, zielony – przykład jest rozwiązany poprawnie.

Moduł programowy Zbiór zadań jest przeznaczony do przechowywania zadań, które użytkownik może rozwiązywać w środowisku rozwiązywania. Aby zacząć rozwiązywanie, należy tylko nacisnąć przycisk *Początek rozwiązywania*.

Środowisko rozwiązywania (ŚR) jest podstawowym modułem ZPM TerM. ŚR jest istotnym narzędziem, przy pomocy którego użytkownik ma możliwość rozwiązywania zadań algebraicznych. Użytkownik może wybrać jedną z trzech opcji rozwiązywania:

- rozwiązywanie automatyczne,
- sprawdzenie kroku rozwiązania albo opcję mieszaną.

Wybermy opcję mieszaną rozwiązywania, która łączy możliwości dwóch pierwszych. Aby zacząć rozwiązywanie, naciśniemy **Start rozwiązywania** na panelu sterowania. Użytkownik może rozwiązać zadanie samodzielnie – wtedy system sprawdza poprawność wykonania transformacji, przy pomocy informatora – użytkownik wybiera transformację, system zaś je wykonuje, oraz za pomocą myszki, wykorzystując metody Drag&Drop oraz DoubleClick.

Wykonajmy pierwszy krok samodzielnie. W tym celu kliknijmy prawym klawiszem myszki na potrzebnym wyrażeniu. Po czym naciśniemy przycisk „Wykonywanie zadania samodzielnie” i wprowadźmy wynik transformacji. Naciskamy przycisk „Sprawdzenie”. System pokazuje czerwone światło. Oznacza to, że transformacji dokonano niepoprawnie. Wprowadźmy poprawne wyrażenie – widzimy, że system daje zielone światło, co świadczy o tym, że transformacja została wykonana prawidłowo.

Korzystając z technologii Drag&Drop, przenieśmy podobne wspólne mnożniki w taki sposób, aby stały one obok siebie. Przy pomocy podwójnego klikania lewym przyciskiem (DoubleClick) można automatycznie dokonać podstawowych czynności.

Aby system automatycznie dokonał transformacji, trzeba nacisnąć prawym przyciskiem myszki na podwyrażenie. Po prawej stronie otworzy się informator (w naszym przypadku otworzył się informator transformacji), w którym należy wybrać transformację do wyrażenia i nacisnąć przycisk „Wykonywanie” (Apply).

Na końcu uproszczenia wyrażenia zapisujemy odpowiedź. Rozwiązane zadania są przechowywane w *Zeszytcie* użytkownika. Każdy użytkownik posiada kilka własnych *Zeszytów*.

Aby rozwiązać zadanie tekstowe, należy najpierw zbudować model matematyczny i nacisnąć przycisk **Wykonywanie (Execute)**. Układ sprawdza poprawność modelu matematycznego.

Budowa poprawnego modelu matematycznego jest pierwszym krokiem do rozwiązywania zadań tekstowych. Dalej należy rozwiązać zwykły układ równań liniowych.

Moduł programowy Informator odgrywa w ZPM TerM bardzo ważną rolę. Zawiera bowiem wszystkie transformacje, z których można korzystać w ŚR, rozwiązując zadanie automatycznie. Każda transformacja zrealizowana została jako informacja zawierająca potrzebne wiadomości. Warto zaznaczyć, że transformacje w informatorze są usystematyzowane.

Rozwiązywacz – klasyczny moduł algebry komputerowej przeznaczony do rozwiązywania (tzn. otrzymania odpowiedzi) typowych zadań, które są przerabiane na lekcjach algebry w 7 klasie szkoły ogólnokształcącej na Ukrainie. Są to:

- równania liniowe oraz układy równań liniowych;
- przeprowadzenie dowodu tożsamości algebraicznych;
- uproszczenie i obliczenie znaczeń wyrażeń algebraicznych.

Moduł programowy Wykresy jest przeznaczony do rozwiązywania zadań w sposób graficzny. Podstawowym zadaniem wśród zadań tego rodzaju jest zadanie rozwiązywania układu 2 równań liniowych z dwiema niewiadomymi. Można tu ustalić punkty oraz linie proste, które są określone przez swoje wzory analityczne, przeprowadzić linię prostą przez dwa punkty oraz znaleźć punkty przecięcia linii prostych, a także otrzymać wzory analityczne zbudowanych obiektów geometrycznych. Przy czym wzory te zawierają ściśle wartości liczbowe.

Perspektywa

Na dzień dzisiejszy system został zrealizowany dla klasy VIII i IX w języku ukraińskim. Obecnie dokonuje się przekładu na język angielski, rosyjski i polski. Trwa przygotowanie TerM 10-11.

Środowisko nauczania na odległość „WebAlmir”

Środowisko nauczania na odległość „WebAlmir” jest pomocą komputerową do kursu algebry liniowej w szkole wyższej.

System „WebAlmir” jest systemem wymagającym logowania: każdy użytkownik ma własną przestrzeń roboczą i możliwość dostępu na odległość do swojego miejsca pracy.



Interfejs środowiska zrealizowano w językach ukraińskim, angielskim i rosyjskim. „WebAlmir” jest zintegrowanym Internet-dodatkiem, na który składają się następujące elementy: *Podręcznik*, *Zbiór zadań*, *Środowisko rozwiązywania zadań*, *Zeszyt*, *Statystyka*, *Dyskusje*, *Generator zadań*. Środowisko jest zabezpieczone zintegrowanym układem bezpieczeństwa, który pozwala na bezproblemowe uzyskiwanie prawa dostępu użytkownika.

Ważnym elementem systemu jest **Podręcznik**. W Podręczniku przechowuje się materiał teoretyczny z kursu algebry liniowej.

Zbiór zadań jest składnikiem systemu, w którym znajdują się wszystkie zadania podtrzymywane środowiskiem rozwiązania.

W **Zeszyt**ie użytkownika są przechowywane rozwiązane zadania.

Element **Środowisko rozwiązania zadań** jest jądrem systemu. Podstawowe funkcje tego modułu to sprawdzanie poprawności transformacji dokonanych przez użytkownika oraz automatyczne dokonanie transformacji na polecenie użytkownika.

Element do prowadzenia statystyki daje możliwość gromadzenia danych oraz przeprowadzania analizy statystycznej w celu stałej kontroli skuteczności nauczania. Na **Forum** istnieje możliwość wymiany zdań i propozycji. Przy pomocy **Generatora zadań** wykładowca może dodawać nowe zadania do Zbioru zadań.

Rozpatrzmy proces rozwiązywania zadania w środowisku. Wybierzmy zadanie w Zbiorze zadań. Naciśnijmy przycisk *Dodać do zeszytu*. Przejdźmy do zeszytu, wybierzmy odpowiednie zadanie i naciśnijmy przycisk *Rozwiązać*. Automatycznie załaduje się środowisko rozwiązywania zadań.

Na środowisko rozwiązywania zadań składają się pole rozwiązywania, pole informacyjne, pole historii wykonanych przez użytkownika czynności, pole komentarza, panel sterowania. Wszystkie czynności w środowisku można wykonać przy pomocy myszki. W polu rozwiązywania znajduje się macierz, nad którą należy dokonać niektórych transformacji. Rozpatrzmy zadanie znalezienia wyznacznika macierzy:

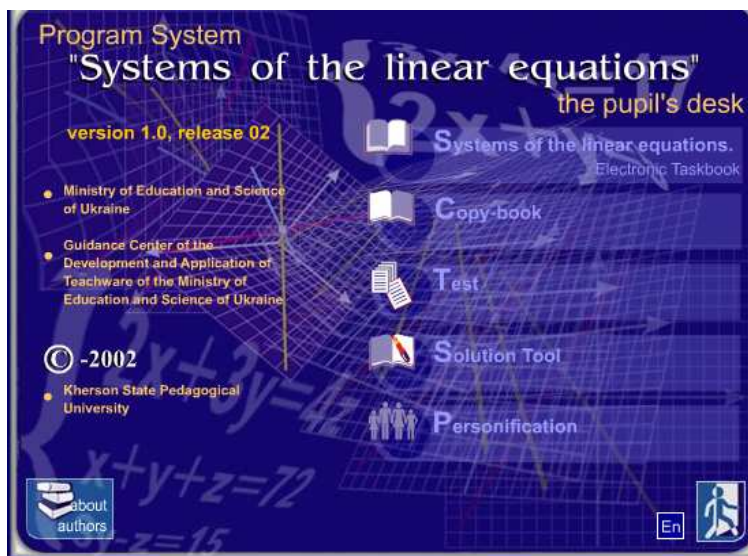
- Zaznaczmy pierwszą linijkę. Pomnóżmy ją przez $-7/4$ oraz dodajmy do drugiej.
- Zaznaczmy pierwszą linijkę. Pomnóżmy ją przez $-3/4$ oraz dodajmy do trzeciej.

W taki sposób kontynuujemy proces, aż doprowadzimy macierz do kształtu przekątnej. Przemnóżmy przez się przekątne elementy oraz określmy wyznacznik. Środowisko automatycznie dokonuje sprawdzenia prawidłowości rozwiązania. Wszystkie czynności użytkownika są zachowywane w historii wykonanych przez użytkownika czynności. Na każdym kroku rozwiązania użytkownik może się zwrócić do podprogramu-eksperta. **Ekspert** dokona następnego kroku z wyjaśnieniem. Cała informacja, z której korzysta użytkownik, jest przechowywana w serwisie www, dzięki czemu istnieje możliwość pracy w środowisku z każdego punktu planety.

Wykorzystywanie segmentowego podejścia oraz zasady architektury otwartej przy budowaniu systemu pozwala abstrahować od pewnej dyscypliny, czyniąc system uniwersalnym. Na podstawie systemu łatwo jest stworzyć wirtualne pracownie z dostępem na odległość.

Układy równań liniowych

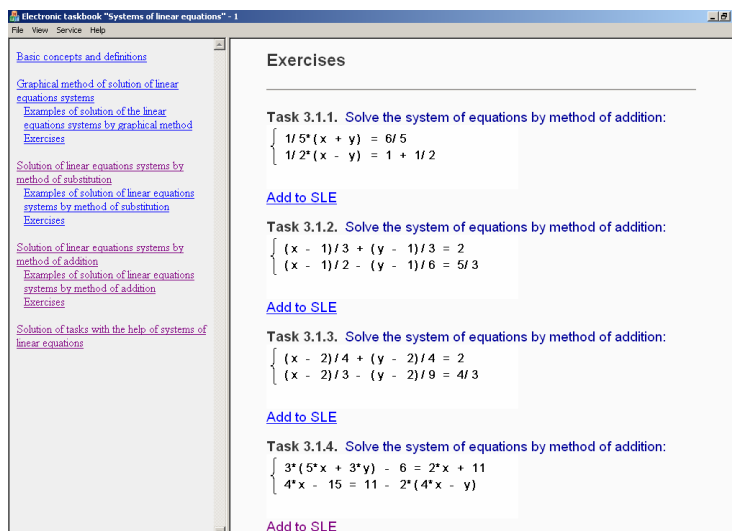
Środowisko programowe „Układy równań liniowych” jest przeznaczone do pomocy przy nauczaniu odpowiedniego tematu z kursu algebry szkoły ogólnokształcącej, a także może być wykorzystane w procesie rozwiązywania zadań arytmetycznych, fizycznych i in., których modelem matematycznym są układy równań liniowych.



Środowisko programowe może być instalowane w dwóch konfiguracjach. Są to *Miejsce pracy ucznia* oraz *Miejsce pracy nauczyciela*. Ich specyficzne moduły różnią się możliwościami funkcyjnymi oraz przeznaczeniem.

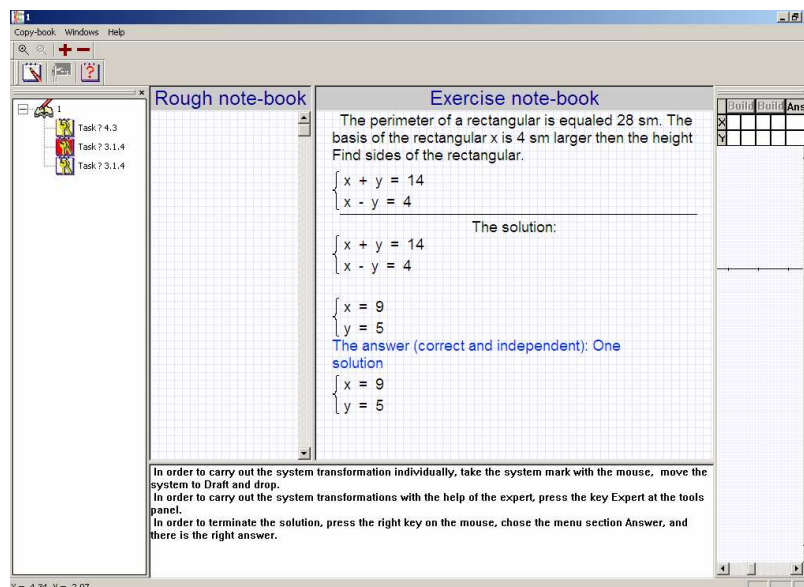
Rozpatrzmy Miejsce pracy ucznia. Składa się ono z następujących modułów: *Logowanie się*, *Środowisko rozwiązywania*, *Testowanie*, *Zeszyt*, *Zbiór zadań i ćwiczeń*.

Praca z programem zaczyna się od procedury zalogowania się: uczeń powinien wybrać swoje imię na liście użytkowników, którą może skorygować przy pomocy odpowiedniego przycisku. Po zalogowaniu się użytkownik uzyskuje dostęp do swojego miejsca pracy.



Treść *Elektronicznego zbioru zadań* jest rozszerzeniem treści odpowiedniego rozdziału szkolnego podręcznika z algebry. Rozdziały zbioru zadań zawierają wiadomości teoretyczne oraz przykłady rozwiązywania zadań w sposób graficzny, przy pomocy metody podstawiania oraz dodawania. Ostatni rozdział zbioru zadań zawiera zadania tekstowe, których modelem matematycznym są układy równań liniowych.

Po naciśnięciu przycisku *Dodać do URL* odbywa się automatyczne załadowywanie modułu „Środowisko rozwiązywania zadań”.



Jest to podstawowy moduł środowiska programowego, ponieważ właśnie on wspiera aktywną działalność matematyczną ucznia. Środowisko rozwiązywania składa się z kilku dziedzin: pole zeszytu, pole brudnopisu, pole czystopisu, pole wykresów.

W celu rozwiązania zadania należy zaznaczyć je w polu zeszytu. Przy tym polecenie zadania automatycznie ukáže się w polu czystopisu. Przenieśmy myszką układ do rozdziału „Rozwiązywanie”. Teraz możemy przejść do bezpośredniego rozwiązywania układu. Na życzenie uczeń może przenieść układ do brudnopisu oraz zachować w czystopisie nie cały tok rozwiązania, tylko niektóre kroki. Przy pomocy myszki dokonujemy transformacji:

- Przenosimy Y z lewej części do prawej
- Podstawiamy X do drugiego równania
- Otwieramy nawiasy
- Sprowadzamy podobne człony
- Znajdujemy Y
- Podstawiamy Y do pierwszego równania
- Znajdujemy X

W efekcie otrzymaliśmy rozwiązanie układu. System automatycznie kontroluje prawidłowość odpowiedzi. Szczególnie interesujące są zadania tekstowe: rozwiązanie takiego zadania wymaga konstruowania przez ucznia modelu zadania matematycznego. Przy rozwiązywaniu zadania tekstowego pojawia się okno, w którym uczeń powinien ułożyć układ równań liniowych, który jest modelem pewnego zadania. Następnie system sprawdza ułożony układ pod kątem prawidłowości i dopiero potem układ jest przenoszony do czystopisu.

Na każdym kroku rozwiązywania układu użytkownik ma możliwość włączenia opcji „Ekspert”, tzn. opcji, w której program dokonuje następnego kroku i wyjaśnia go.

System testowania pozwala na sprawdzenie znajomości materiału teoretycznego. System automatycznie wybiera 1 z 6 wersji zadań. Podsumowanie testowania pozwala na ścisłe ocenianie wiedzy ucznia z każdego zagadnienia.

Generator zadań jest głównym modułem miejsca pracy nauczyciela. Przy pomocy danego modułu nauczyciel ma możliwość sformułowania zadania dla uczniów oraz przesłania go do miejsca pracy ucznia. Po rozwiązaniu zadania przez ucznia nauczyciel ma możliwość sprawdzić wyniki rozwiązania.

ŚP otrzymało rekomendację Ministerstwa Oświaty i Nauki Ukrainy oraz certyfikat zgodności ze standardami UkrNCSoftrating. Wersja demonstracyjna ŚP jest dostępna na stronie UCh.

Bibliografia:

1. <http://ksu.edu.ua>
2. <http://krug.kspu.ks.ua>