

ЕЛЕКТРОННИЙ КОНСТРУКТОР СТЕРЕОМЕТРИЧНИХ ЗАДАЧ

Калугін Руслан

Науковий керівник: д-р. пед. наук, професор Лов'янова І. В.

Криворізький державний педагогічний університет,

м. Кривий Ріг, Україна

У статті представлено веб-орієнтовану розробку конструктора стереометричних задач, призначеного для організації практичних занять магістрів спеціальності 014 Середня освіта (Математика). Розроблений програмний засіб дає змогу автоматично формувати тексти задач з тем «Правильна трикутна призма», «Правильна чотирикутна призма», «Правильна трикутна піраміда», «Правильна чотирикутна піраміда» відповідно до обраних користувачем відомих та шуканих вимірів та величин. Електронний конструктор стане одним з різновидів навчального контенту онлайн-курсу «Розвиток логічного мислення старшокласників у навчанні математики», спрямованого, зокрема, на вироблення умінь здійснювати аналітико-синтетичні міркування.

Ключові слова: стереометрична задача, конструктор, онлайн-курс, аналітико-синтетичні міркування.

Electronic constructor of stereometric problems

R. Kaluhin

Scientific supervisor: Doctor of Pedagogical Sciences, Professor Lovianova I. V.

Kryvyi Rih State Pedagogical University,

Kryvyi Rih, Ukraine

The article presents a web-oriented constructor of stereometric problems, intended for the organization of practical classes for master's students majoring in the specialty 014 Secondary education (Mathematics). This software makes it possible to automatically generate the texts of problems on the topics "Regular Triangular Prism", "Regular Quadrangular Prism", "Regular Triangular Pyramid", "Regular Quadrangular Pyramid" in accordance with the known and unknown measurements and quantities selected by the user. The electronic constructor will be one of the types of educational content in the online course "Development of logical thinking of high school students in teaching Mathematics", aimed in particular at developing the ability to analytical and synthetic reasonings.

Key words: stereometric problem, constructor, online course, analytical and synthetic reasonings.

Постановка проблеми. Сучасна освітня політика вітчизняних ЗВО передбачає можливість проведення занять у різних форматах: аудиторне навчання, дистанційне навчання, змішане навчання тощо. Усі ці форми організації освітнього процесу потребують розробки якісних навчальних матеріалів, зручних для використання викладачем та студентом будь-де і будь-коли. Найпоширенішим засобом систематизації таких матеріалів та презентації

їх широкому колу осіб, як відомо, є онлайн-курс. І якщо в умовах аудиторного навчання онлайн-курс є бажаним, але не обов'язковим, то змішане і дистанційне навчання без онлайн-курсів відповідних навчальних дисциплін організувати практично неможливо.

Спектр різновидів навчального контенту, який може містити онлайн-курс, досить широкий і залежить, насамперед, від характеру навчальної дисципліни, авторського задуму та практичного досвіду викладання, що має розробник онлайн-курсу, проте основними елементами онлайн-курсів традиційно вважають матеріали лекцій (у формі текстових файлів чи відео-фрагментів), завдання для практичних занять, тренувальні та контрольні тести, форуми для спілкування з викладачем та однокурсниками тощо.

Узагальнюючи вітчизняний досвід упровадження онлайн-освіти та викладання математики й інформатики зокрема, S. Shokalyuk та ін. [6] виділяють три основні підходи до реалізації дистанційної підтримки освітнього процесу: 1) розміщення навчальних ресурсів та навчально-методичних матеріалів на офіційних веб-сторінках закладу освіти, його структурного підрозділу або ж на персональних веб-сайтах і блогах викладача – переважно з використанням хмарних сховищ (наприклад, *Google Диска*); 2) презентація сукупності освітніх ресурсів як електронних курсів, створених за допомогою систем управління навчанням (наприклад, *Moodle* чи *Google Classroom*); 3) використання спеціалізованих онлайн-інструментів (як-от *CoCalc*, *Jupyter Notebook* тощо).

Аналіз досліджень і публікацій. У працях вітчизняних та зарубіжних науковців окреслено різні аспекти й проблеми онлайн-освіти та розроблення університетських онлайн-курсів. Так, J. Gaytan і В. McEwen [2] досліджують оцінювальні стратегії як одні з найефективніших в онлайн-навчанні; М. Dixon [3] вивчає питання мотивації студентів, залучених до проходження онлайн-курсів і виокремлює з-поміж факторів, що впливають на успішність здобувачів освіти, орієнтованість програми онлайн-курсу на взаємодію студентів між собою та наявність у ньому інтерактивних, практичних і творчих завдань; Р. Holzweiss та ін. [4] порушують проблему тайм-менеджменту студентів в

умовах дистанційного навчання. Дослідження К. Vlasenko та ін. [5] стосується онлайн-освіти магістрів спеціальності «Математика» і доводить ефективність використання в онлайн-курсі завдань різних типів, а також опорних схем, динамічних графіків і анімованих зображень.

Викладачі курсу *Get Interactive: Practical Teaching with Technology* [1], розміщеного на платформі МВОК *Coursera*, дають слушні поради щодо включення в онлайн-курс елементів інтерактивної взаємодії, що дають змогу поживити освітній процес та урізноманітнити форми роботи зі студентами. Власне, навчання на цьому курсі спонукало автора до розроблення інтерактивних ресурсів для майбутнього онлайн-курсу «Розвиток логічного мислення старшокласників у навчанні математики», робота над яким триває.

Одна з тем згаданого курсу присвячена виробленню умінь здійснювати аналітико-синтетичні міркування у ході розв'язування планіметричних та стереометричних задач. Відтак, виникла потреба в інструменті для створення пакету завдань, які можна було б використовувати як засіб відпрацювання процедури проведення логіко-математичного аналізу тексту задачі та для організації самостійної роботи студентів з розв'язування задач.

Метою статті є презентація авторського електронного конструктора стереометричних задач як елемента онлайн-курсу для магістрів спеціальності 014 Середня освіта (Математика).

Виклад основного матеріалу (результатів) дослідження. Опишемо процес реалізації задуму цього програмного засобу за такими напрямками: 1) тематична спрямованість задач; 2) принцип роботи конструктора; 3) логіка програмного коду та інтерфейс користувача; 4) приклади задач, згенерованих конструктором.

Тематична спрямованість задач. Відправною точкою у роботі над конструктором став вибір тематики задач, тексти яких можна було б згенерувати автоматично. На етапі проєктування за основу було взято тему «Правильна трикутна піраміда», проте у процесі програмування конструктора перелік тем було розширено. Так, поточна версія конструктора призначена для

генерації задач на правильні трикутну і чотирикутну призми, правильні трикутну і чотирикутну піраміди.

Принцип роботи конструктора. Аналізуючи тексти стереометричних задач, представлених в шкільних підручниках з математики, помічаємо, що найчастіше задача на правильну призму чи піраміду містить два відомі елементи, один з яких належить площині основи, а інший – поза цією площиною. Саме така комбінація відомих вимірів однозначно задає многогранник, основою якого є правильний многокутник. Так, наприклад, у випадку правильної трикутної піраміди, знаючи хоч один елемент трикутника-основи, можна знайти інші його елементи і використати їх для того, щоби «поєднати» їх з вимірами піраміди, які не належать площині основи.

Розробляючи конструктор, ми обрали такі виміри правильної трикутної піраміди:

– *лінійні*: висота піраміди; бічне ребро піраміди; апофема піраміди; відстань від центра основи піраміди до її бічного ребра; відстань від центра основи піраміди до її бічної грані (*група L_1*); сторона основи піраміди; висота основи піраміди; радіус кола, вписаного в основу піраміди; радіус кола, описаного навколо основи піраміди (*група L_2*);

– *кутові*: кут нахилу бічних ребер піраміди; кут нахилу бічних граней піраміди; плоский кут при вершині піраміди; кут між висотою і бічним ребром піраміди; кут між бічним ребром і стороною основи піраміди (*група K*).

Конструктором передбачено генерацію задач двох типів: з відомими двома лінійними вимірами (один – з *групи L_1* , інший – з *групи L_2*), з відомими лінійним (з *групи L_1* чи *групи L_2*) та кутовим (з *групи K*) вимірами.

Набір елементів для задач на правильну чотирикутну піраміду обирався аналогічно, за винятком елемента *висота основи* в *групі L_2* , який замінено на *діагональ основи*.

Правильні трикутна та чотирикутна призми мають свої специфічні елементи *групи L_1* (діагональ бічної грані призми; відстань від центра нижньої основи призми до сторони її верхньої основи; відстань від центра нижньої

основи призми до вершини її верхньої основи) та *групи K* (наприклад, кут нахилу діагоналі бічної грані призми до площини її основи, кут між діагоналями сусідніх бічних граней – для трикутної та чотирикутної призми; кут нахилу діагоналі призми до площини її основи – для чотирикутної призми).

У конструкторі передбачено також вибір типу майбутньої задачі за шуканими даними (на знаходження якого-небудь лінійного чи кутового виміру; на знаходження площі бічної поверхні чи площі повної поверхні; на знаходження об'єму).

Логіка програмного коду та інтерфейс користувача. Розроблений інтерфейс конструктора значною мірою впливає з моделі конструктора, представленої на рис. 1.

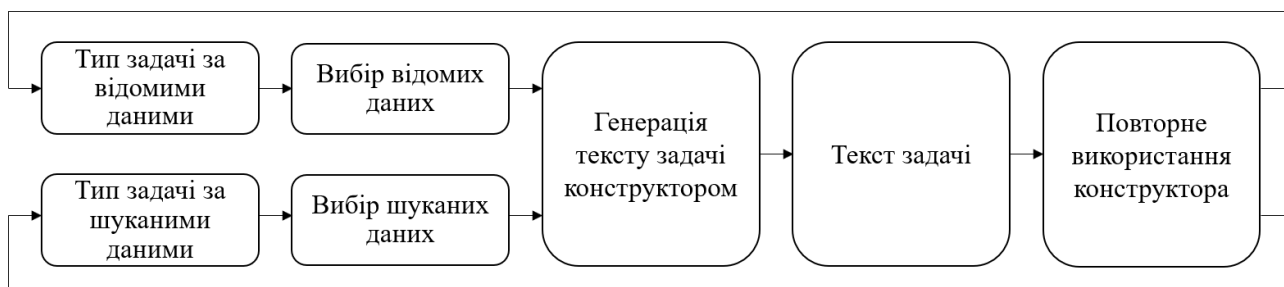


Рис. 1. Модель конструктора стереометричних задач

Відповідно до цієї моделі та загального принципу роботи бажаного інструмента, підготовлено програмний код веб-орієнтованого конструктора, який можна вбудувати в онлайн-курс. Основним елементом конструктора є форма, на якій розміщено розкриті списки та пояснювальні надписи до них, кнопка генерації тексту задачі, посилання на головну форму (меню) конструктора. Отож, конструктор є звичайною HTML-сторінкою з елементами *form, label, select, button*.

У якості прикладу наведемо фрагмент підготовленого HTML-коду сторінки конструктора задач на правильну трикутну піраміду, що відповідає створенню розкритого списку елементів *групи L₁* (див. рис. 2).

```

<li id="li_3" >
  <label id="label_3" hidden="true">Відомий перший лінійний вимір </label>
  <div>
    <select class="element select large" id="element_3" name="element_3" hidden="true">
      <option value="" selected="selected"></option>
      <option value="1" >Висота піраміди</option>
      <option value="2" >Бічне ребро піраміди</option>
      <option value="3" >Апофема піраміди</option>
      <option value="4" >Відстань від центра основи піраміди до її бічного ребра</option>
      <option value="5" >Відстань від центра основи піраміди до її бічної грані</option>
    </select>
  </div>
</li>

```

Рис. 2. Фрагмент HTML-коду розробленого конструктора

Функціональність конструктора забезпечено обробниками подій «зміна розкритого списку» (зокрема, для динамічного представлення списків з відомими та шуканими елементами відповідно до обраного типу задачі), «натиснення на кнопку» (для перевірки на коректність вибору відомих та шуканих елементів та для генерації тексту задачі).

Текст задачі формується конструктором поступово, відповідно до номерів обраних елементів розкритих списків, і записується у рядкову змінну, значення якої виводиться на екран. Фрагмент JS-коду конструктора, що ілюструє забезпечення граматичної коректності майбутнього тексту задачі, наведено на рис. 3.

```

case 2:
  switch(element_5.selectedIndex){
    case 1: k = k + 'Висота правильної трикутної піраміди дорівнює H'; break;
    case 2: k = k + 'Бічне ребро правильної трикутної піраміди дорівнює b'; break;
    case 3: k = k + 'Апофема правильної трикутної піраміди дорівнює l'; break;
    case 4: k = k + 'Відстань від центра основи правильної трикутної піраміди до її бічного ребра дорівнює p'; break;
    case 5: k = k + 'Відстань від центра основи правильної трикутної піраміди до її бічної грані дорівнює q'; break;
    case 6: k = k + 'Сторона основи правильної трикутної піраміди дорівнює a'; break;
    case 7: k = k + 'Висота основи правильної трикутної піраміди дорівнює h'; break;
    case 8: k = k + 'Радіус кола, вписаного в основу правильної трикутної піраміди, дорівнює r'; break;
    case 9: k = k + 'Радіус кола, описаного навколо основи правильної трикутної піраміди, дорівнює R'; break;
  }
  switch(element_6.selectedIndex){
    case 1:
      if(element_5.selectedIndex == 2) k = k + ' і нахилене до площини основи під кутом  $\alpha$ . ';
      else k = k + '. Бічні ребра піраміди нахилені до площини основи під кутом  $\alpha$ . ';
      break;
    case 2: k = k + '. Бічні грані піраміди нахилені до площини основи під кутом  $\beta$ . '; break;
    case 3: k = k + '. Плоский кут при вершині піраміди дорівнює  $\gamma$ . '; break;
    case 4:
      if (element_5.selectedIndex == 1) k = k + ' і утворює з її бічним ребром кут  $\phi$ . ';
      else if (element_5.selectedIndex == 2) k = k + ' і утворює з її висотою кут  $\phi$ . ';
      else k = k + '. Висота і бічне ребро піраміди утворюють кут  $\phi$ . ';
      break;
    case 5:
      if (element_5.selectedIndex == 6) k = k + ' і утворює з її бічним ребром кут  $\omega$ . ';
      else if (element_5.selectedIndex == 2) k = k + ' і утворює зі стороною її основи кут  $\omega$ . ';
      else k = k + '. Бічне ребро піраміди і сторона її основи утворюють кут  $\omega$ . ';
      break;
  }
}

```

Рис. 3. Фрагмент JS-коду конструктора

Для публікації конструктора в мережі використано сервіс для спільної розробки програмного забезпечення *GitHub*. Файли проєкту розміщено у

спеціально створеному для цього репозитарії, доступ до якого відкрито за посиланням: <https://r-kalugin.github.io/Constructor/>.

Загальний вигляд окремої сторінки конструктора представлено на рис. 4.

Конструктор задач
з теми «Правильна трикутна піраміда»

Тип задачі за відомими даними

Відомий лінійний та кутовий виміри ▼

Тип задачі за шуканими даними

На знаходження об'єму ▼

Відомий лінійний вимір

Висота піраміди ▼

Відомий кутовий вимір

Кут між висотою і бічним ребром піраміди ▼

Висота правильної трикутної піраміди дорівнює H і утворює з її бічним ребром кут φ . Знайдіть об'єм цієї піраміди.

[Повернутись до головного меню](#)

Рис. 4. Сторінка конструктора

Подамо *приклад* задач, тексти яких підготовлено засобами розробленого конструктора:

1. Діагональ бічної грані правильної трикутної призми дорівнює d , а висота основи цієї призми дорівнює h . Знайдіть площу бічної поверхні призми.

2. Бічне ребро правильної трикутної призми дорівнює b , а сторона основи цієї призми дорівнює a . Знайдіть відстань від центра нижньої основи призми до вершини її верхньої основи.

3. Радіус кола, вписаного в основу правильної трикутної призми, дорівнює r . Кут між діагоналями сусідніх бічних граней призми дорівнює ω . Знайдіть бічне ребро цієї призми.

4. Діагональ правильної чотирикутної призми дорівнює m , а діагональ основи цієї призми дорівнює c . Знайдіть площу повної поверхні призми.

5. Діагональ правильної чотирикутної призми дорівнює m і нахилена до площини її основи під кутом β . Знайдіть кут між діагоналями сусідніх бічних граней призми.

6. Бічне ребро правильної чотирикутної призми дорівнює b і утворює з діагоналлю її бічної грані кут γ . Знайдіть об'єм цієї призми.

7. Висота правильної трикутної піраміди дорівнює H і утворює з її бічним ребром кут φ . Знайдіть об'єм цієї піраміди.

8. Апофема правильної трикутної піраміди дорівнює l , а радіус кола, описаного навколо основи цієї піраміди, дорівнює R . Знайдіть плоский кут при вершині піраміди.

9. Висота правильної трикутної піраміди дорівнює H , а радіус кола, описаного навколо основи цієї піраміди, дорівнює R . Знайдіть відстань від центра основи піраміди до її бічного ребра.

10. Апофема правильної чотирикутної піраміди дорівнює l , а діагональ основи цієї піраміди дорівнює c . Знайдіть відстань від центра основи піраміди до її бічної грані.

11. Відстань від центра основи правильної чотирикутної піраміди до її бічного ребра дорівнює p , а діагональ основи цієї піраміди дорівнює c . Знайдіть кут нахилу бічних ребер піраміди до площини її основи.

12. Радіус кола, вписаного в основу правильної чотирикутної піраміди, дорівнює r . Плоский кут при вершині піраміди дорівнює γ . Знайдіть площу бічної поверхні цієї піраміди.

Висновки. Підсумовуючи зауважимо, що практична цінність розробленого конструктора полягає в можливості генерації задач різних рівнів складності. Оскільки вміння оцінити рівень складності математичної задачі – це

одна зі складових методичної компетентності майбутнього вчителя математики, в онлайн-курсі, згаданому вище, одне з обов'язкових оцінюваних практичних завдань буде сформульовано так: «Використовуючи конструктор задач, посилання на який подано в рубриці “Ресурси теми”, сформууйте три задачі різних рівнів складності. Розв'яжіть ці задачі».

Перспективи подальших досліджень вбачаємо у наповненні розробленого конструктора іншими темами шкільного курсу стереометрії та в підготовці збірника задач, згенерованих конструктором.

Список використаної літератури

1. Get Interactive: Practical Teaching with Technology. URL: <https://www.coursera.org/learn/getinmooc>.
2. Gaytan, J. and McEwen, B. Effective Online Instructional and Assessment Strategies. *The American Journal of Distance Education*. 2007. 21 (3). PP. 117–132. URL: <https://doi.org/10.1080/08923640701341653>.
3. Dixon, M. Creating effective student engagement in online courses: What do students find engaging? *Journal of the Scholarship of Teaching and Learning*. 2010. 10(2). PP. 1–13. URL: <https://eric.ed.gov/?id=EJ890707>.
4. Holzweiss, P. and Polnick, B. and Lunenburg, F. Online in half the Time: a Case Study with Online Compressed Courses. *Innovative Higher Education*. 2019. 44(4). PP. 299–315. URL: <https://eric.ed.gov/?id=EJ1222651>.
5. Vlasenko, K., Lovianova, I., Chumak, O., Sitak, I. and Achkan, V. The arrangement of on-line training of master students, majoring in Mathematics for internship in technical universities. *Journal of Physics: Conference Series*. 1840 (2021). P. 012007. URL: <https://doi.org/10.1088/1742-6596/1840/1/012007>.
6. Shokalyuk, S., Bohunenko, Y., Lovianova, I. and Shyshkina, M. Technologies of distance learning for programming basics lessons on the principles of integrated development of key competences. *Proceedings of the 7th Workshop on Cloud Technologies in Education (CTE 2019)*. 2020. Vol. 2643. PP. 548–562. URL: <http://ceur-ws.org/Vol-2643/paper32.pdf>.