



Вікторія МУЛИК,
учителька інформатики Кам'янець-Подільського
ліцею №5, учитель-методист

Віталій РЕБРИНА,
методист НМЦ навчально-інформаційного
забезпечення, ІКТ і видавничої діяльності
ХОІППО ім.А.Назаренка, відмінник освіти України

Роль компетентнісних задач при вивченні теми «Алгоритмізація та програмування» у 8 класі

Компетентнісні задачі відіграють ключову дидактичну та методичну роль у вивченні теми «Алгоритмізація і програмування» у 8 класі, оскільки забезпечують перехід від формального засвоєння синтаксису до усвідомленого застосування алгоритмічного мислення в практичних ситуаціях.

Основні аспекти ролі компетентнісних задач

1. Формування алгоритмічного та логічного мислення
Компетентнісні задачі:

- орієнтують учнів на аналіз проблемної ситуації, а не на відтворення готових алгоритмів;
- вимагають декомпозиції задачі на підзадачі;
- сприяють усвідомленню понять умови, циклу, змінної, розгалуження як інструментів розв'язання реальних проблем.

Отже, алгоритм постає не як абстрактна схема, а як засіб досягнення мети.

2. Зв'язок програмування з реальним життям

На відміну від тренувальних вправ на кшталт «вивести числа від 1 до 10», компетентнісні задачі:

- моделюють побутові, навчальні або соціальні ситуації (підрахунок витрат, планування маршруту, обробка результатів вимірювань тощо);
- демонструють практичну цінність програмування;
- підвищують навчальну мотивацію восьмикласників.

Учні починають сприймати програмування як корисний інструмент, а не лише як шкільний предмет.

3. Розвиток ключових і предметних компетентностей

Компетентнісні задачі сприяють одночасному формуванню:

- інформаційно-цифрової компетентності (робота з даними, алгоритмами, програмами);
- математичної компетентності (обчислення, залежності, логічні операції);

- уміння вчитися (планування дій, пошук помилок, рефлексія);
- ініціативності та підприємливості (оцінювання варіантів рішень, оптимізація).

4. Диференціація та індивідуалізація навчання
Компетентнісні задачі легко варіюються:

- за рівнем складності;
- за кількістю умов;
- за способом реалізації (різні алгоритмічні підходи).
Це дозволяє:
- залучити до роботи учнів з різним рівнем підготовки;
- організувати проектну та групову діяльність;
- підтримувати інтерес як сильних, так і слабших учнів.

5. Формування навичок аналізу та налагодження програм
Оскільки компетентнісні задачі зазвичай не мають єдиного «шаблонного» розв'язку, учні:

- навчаються перевіряти правильність алгоритму;
- аналізувати помилки;
- удосконалювати програму.

Це наближає навчальний процес до реальної практики програмування.

Авторські колективи підручників з інформатики для 8 класу підібрали пакет компетентнісних задач, частину яких пропонуємо у цій статті.

Наведемо окремі приклади компетентнісних задач з чинних підручників з інформатики для 8 класу.

Задача. Напишіть програму для визначення, скільки квадратних метрів зелених насаджень потрібно населеному пункту, де ви проживаєте, якщо за нормами Всесвітньої організації охорони здоров'я площа озеленення міст має становити 21 м² зелених насаджень на одну людину.

Розв'язок мовою програмування Python може бути такий:

```
n=int(input('Скільки осіб проживає у Вашому населеному пункті?'))  
print(n*21)
```

Задача. Відомі назва планети, радіус орбіти, швидкість руху по орбіті. Обчислити тривалість року на планеті (період обертання). Перевірити роботу програми на тестових даних.

Назва планети	Радіус орбіти, км	Швидкість руху, км/год	Період обертання, діб
Земля	149 600 000	107 250	365,18
Венера	108 200 000	126 110	
Нептун	4 496 600 000	19 550	
Марс	227 900 000	86 870	

Розв'язок мовою програмування Python може буде такий:

```
from math import pi
s=input('назва планети ')
r=int(input('радіус орбіти '))
v=int(input('швидкість руху по орбіті '))
r=r*1000
v=v*1000/3600
T=(2*pi*r)/v
T=T/(3600*24)
print('%.2f%%T')
```

Задача. Відомо, що в результаті переробки 60 кг макулатури зберігається 1 дерево і 0,0005 га лісу. Скласти програму Збережи дерево, яка за масою макулатури визначає, скільки дерев і яку площу лісу допоможе зберегти її переробка. Розробити консольну і графічну програми.

Консольна програма:

```
n=float(input('Введи масу макулатури '))
d=int(n)//60
g=0.0005*n/60
print('Збереже дерев',d)
print('Збереже лісу','%.4f%%g,'ra')
```

Графічна програма:

```
def f1():
    s=e.get()
    if s=="":
        pass
    else:
        s1=int(s)
        k=s1//60
        l1=Label(w,text='Збережено дерев '+str(k),
font='Arial 14',fg='red')
        l1.place(x=20,y=180)
        s2=float(s)
        g=s2*0.0005
        l2=Label(w,text='Збережено лісу '+str(g)+"га",
font='Arial 14',fg='red')
        l2.place(x=20,y=220)
    from tkinter import *
    w=Tk()
    w.geometry('300x300+400+300')
    w.title('Збережи дерево')
    w.resizable(0,0)
    l=Label(w,text='Введіть масу макулатури', font='Arial 14')
    l.place(x=10,y=10)
    e=Entry(w,width=10, font='Arial 14')
    e.place(x=70,y=50)
    b=Button(w, text='Розрахувати', font='Arial
```

```
14',command=f1)
b.place(x=100,y=100)
w.mainloop()
```

Задача. Скласти програму Касир, під час виконання якої комп'ютер буде виконувати такі дії:

- повідомляти вартість одиниці товару;
- запитувати кількість товару, що купується;
- обчислювати вартість покупки;
- повідомляти вартість покупки;
- запитувати, скільки грошей дає покупець;
- повідомляти решту.

```
def f1():
    global e1,k
    s=e.get()
    if s=="":
        pass
    else:
        s1=int(s)
        k=s1*134
        l1=Label(w,text='Ваша вартість покупки '+str(k),
font='Arial 14',fg='#660000')
        l1.place(x=20,y=200)
```

```
        l2=Label(w,text='Скільки грошей Ви даєте? ',
font='Arial 14',fg='#660000')
        l2.place(x=20,y=250)
        e1=Entry(w,width=10, font='Arial 14')
        e1.place(x=300,y=250)
        b1=Button(w,text='Ваша решта',font='Arial
14',command=f2)
        b1.place(x=200,y=300)
```

```
def f2():
    r=e1.get()
    if r=="":
        pass
    else:
        t=int(r)-k
        l5=Label(w,text='Ваша решта '+str(t), font='Arial
14',fg='#660000')
        l5.place(x=20,y=350)
```

```
from tkinter import *
```

```
w=Tk()
w.geometry('500x500+400+200')
w.title('касир')
w.resizable(0,0)
```

```
l=Label(w,text='Одиниця товару коштує 134 грн',
font='Arial 14',fg='red')
l.place(x=100,y=10)
l3=Label(w,text='Введіть кількість одиниць, яку будете
купувати', font='Arial 14')
l3.place(x=10,y=50)
e=Entry(w,width=10, font='Arial 14')
e.place(x=180,y=100)
b=Button(w,text='Обчислити вартість покупки',font='Arial
14',command=f1)
b.place(x=100,y=150)
w.mainloop()
```

Задача. Обчислити значення щомісячної виплати m за відомими значеннями суми позики s , відсотка p і кількості років n за формулою

$$m = \frac{sr(1+r)^n}{12((1+r)^n - 1)}, \text{ де } r = \frac{p}{100}$$

```
s=float(input("Введи суму позики "))
p=float(input("Введи відсотки "))
n=int(input("Введи кількість років "))
r=p/100
m=(s*r*(1+r)**n)/(12*((1+r)**n-1))
print("Щомісячна виплата буде", "%2f" % m, 'грн')
```

Отже, у темі «Алгоритмізація і програмування» у 8 класі компетентнісні задачі виконують роль:

- містка між теорією та практикою;
- засобу розвитку алгоритмічного мислення;
- інструменту формування ключових компетентностей;
- мотиваційного чинника навчання.

Саме через такі задачі програмування стає не самоціллю, а ефективним засобом розв'язання життєвих і навчальних проблем.

Використані джерела

1. Інформатика : підруч. для 8 кл. закл. загал. серед. освіти / О.О.Бондаренко, В.В.Ластовецький, О.П.Пилипчук, Є.А.Шестопалов. - Харків: Вид-во "Ранок", 2025. - 272 с.: іл.
2. Інформатика : підруч. для 8 кл. закл. загал. серед. освіти / Й.Ривкінд та ін. - Київ: Генеза, 2025. - 304 с.: іл.
3. Інформатика : підруч. для 8 кл. закл. загал. серед. освіти / О.В.Коршунова, І.О.Завадський. - К.: Видавничий дім "Освіта", 2025. - 256 с.: іл.
4. Інформатика : підруч. для 8 кл. закл. загал. серед. освіти / Джон Ендрю Біос. - К.: Видавництво "Лінгвіст", 2025. - 208 с.: іл.
5. Інформатика : підруч. для 8 кл. закл. загал. серед. освіти / Н.В.Морзе, О.В.Барна. - Київ: УОВЦ "Оріон", 2025 - 240 с.: іл.
6. Дистанційна школа програмування. Web-ресурс. <https://sbs.hoippo.km.ua/>



Віталій ЗУБИК,
методист НМЦ навчально-інформаційного забезпечення, інформаційно-комунікаційних технологій та видавничої діяльності ХОІППО ім.А.Назаренка

Система DMOJ – інструмент вчителя для роботи з обдарованими учнями

Система на <http://dn.hoippo.km.ua:8885/> — це автоматизована система перевірки завдань з програмування, розгорнута на базі DMOJ, яка використовується Хмельницьким обласним інститутом післядипломної педагогічної освіти для організації олімпіад, відбіркових турів до Всеукраїнської олімпіади з інформатики, заочних змагань та тренувань учнів Хмельницької області.

Вона спеціально адаптована для потреб учителів інформатики Хмельниччини і є одним із основних інструментів підготовки школярів до обласних та всеукраїнських змагань та роботи з обдарованими учнями.

Так, система DMOJ не просто підходить для роботи з обдарованими учнями, а є одним із найкращих інструментів для їхньої підготовки до олімпіад з інформатики та розвитку алгоритмічного мислення. Її архітектура та функціонал орієнтовані на підтримку учнів з високим рівнем мотивації та здібностей.

Обдаровані учні швидко "переростають" шкільну програму. Система надає доступ до задач, які за складністю відповідають рівням змагань різного ступеня:

- Задачі олімпіадного рівня: Система містить велику бібліотеку задач, які охоплюють усі теми, необхідні для підготовки до Всеукраїнської олімпіади з інформатики. Це задачі на динамічне програмування, графи, потокові алгоритми, геометрію, структури даних (наприклад, дерево відрізків, SQRT-декомпозиція).

- Підтримка мов програмування та алгоритмів: Система підтримує сучасні стандарти мов програмування (C++17/20, Java, Python та інші (всього близько 16 на нашому сервері)), що дозволяє учням використовувати всі переваги мови програмування для написання ефективного коду.

- Багаторівнева система оцінювання: Задачі мають "вагу" (points), яка приблизно відповідає їхній складності. Учень, який розв'язує складну задачу на 50 балів, отримує значно більший приріст у рейтингу, ніж той, хто розв'язує багато легких задач, що стимулює учня братися за дійсно складні задачі.