

УДК 374.31:004(07)

Косінова Надія

ВИКОРИСТАННЯ ПЕРЕЛІКОВНИХ ТИПІВ ПРИ РОЗВ'ЯЗУВАННІ ЗАДАЧ НА МОВІ ПАСКАЛЬ

Пропонується набір задач, розв'язаних на мові Паскаль, в яких використані дані перелікового типу. Використання даних перелікового типу спрощує розв'язування цілого класу типових задач при вивченні програмування. Кожна мова програмування має засоби для розв'язування таких задач. В основі лежить принцип «Краще один раз побачити ніж сто раз почути». Приведені готові програми, в яких зроблені акценти на використання перелікового типу та приведені теоретичні викладки про переліковий тип в мові Paskal. Розрахована для осіб, що вивчають мову Паскаль або викладають її.

Ключові слова: *STEM-спеціальність, переліковий тип, мова програмування, програма, дані, масив даних.*

Постановка проблеми. Одним з основних напрямів оновлення змісту освіти є забезпечення її якості на основі новітніх досягнень науки, культури і соціальної практики. Нині одною із перспективних освітніх стратегій вважається STEM (Science, Technology, Engineering, Math) – послідовність курсів або програм навчання, з використанням яких здійснюється підготовка учнів до здобуття хорошої освіти та успішного працевлаштування. STEM-освіта є пріоритетною з причин затребуваності ІТ-фахівців – програмістів, інженерів, фахівців технологічних виробництв, спеціалісти біо-, нано-, конго-, інфо- технологій [2].

Здобуття таких професій потребує всебічної підготовки в різних освітніх областях природничих наук, інженерії, технологій та програмування, напрямів, які охоплює STEM-освіта. Програмування є основою для здобуття професії ІТ-спеціаліста.

Навчання цій професії потребує різних технічно складних навичок із застосуванням математичних знань і наукових понять. Учням потрібно розвивати логічне мислення та технічну грамотність щоб навчитися вирішувати проблеми та стати новаторами, винахідниками.

Навчитися вирішувати проблеми можна тільки з допомогою практики. В програмуванні такою практикою є розв'язування задач. Набір задач на одну тему дає можливість вивчити всі тонкості її, одночасно повторюючи оператори попередніх тем.

Аналіз останніх публікацій і досліджень. Дійсно, аналіз інформаційних і літературних джерел вказує на те, що напрям STEM, у т.ч. і підготовку фахівців у цьому векторі досліджують у своєму доробку такі дослідники, як: І. Василяшко [11, с. 22–24], О. Гермак [6, с. 86–91], Н. Гончарова [11, с. 25–27; 12, с. 30–33], Г. Єльнікова [3], О. Кузьменко [10, с. 72–74; 11, с. 59–62; 12, с. 69–70], О. Патрикеєва [10, с. 81–83; 11, с. 74–76], М. Ростока [7, с. 60–67; 12, с. 84–86], І. Савченко [7, с. 47–60; 12, с. 117–120], а також Н. Поліхун, І. Сліпухіна, О. Стрижак і І. Чернецький [6, с. 50–62; 7, с. 88–103] тощо.

Проте, досить актуальним питанням у цьому сенсі залишається проблема підготовки STEM-спеціалістів з програмування, зокрема їхнє навчання мовам програмування, у т.ч. і мові «Паскаль». Адже, підготовці ІТ-спеціалістів з програмування, зокрема на мові Паскаль, а саме навчання до використання перелікового типу в Паскалі присвячено чимало робіт дослідників, а саме таких, як: Ю. Бородич, А. Вальвачев і А. Кузьмич [1]; Т. Караванова [4]; Н.Культин [5], В. Пильщиков [8], В. Сікора і І. Юрченко [14; 15]; В. Фаронов [13] та ін.

Проте, застосування цих прийомів і методів авторами не достатньо висвітлюється у практичному аспекті, зокрема при розв'язуванні певних задач, що є важливим для мотивування учнів у вивченні мов програмування.

Формулювання мети і завдань статті. У форматі STEM-освіти задачі наведені у термінах реального світу і підібрані в порядку ускладнення, щоб відобразити всі нюанси використання даних перелікового типу, тобто програми,

що реалізують дані задачі створені та відлагоджені в середовищі системи Turbo-pascal 5.5. Кожна задача мотивує до використання перелікового типу. У програмі кольором відмічені місця його використання та надані, при необхідності, відповідні пояснення окремими написами або виділеннями за допомогою курсивного шрифту.

Тому, **метою статті** є визначення практичної необхідності використання перелікового типу даних при програмуванні на мові Паскаль шляхом наведення прикладів розв'язку 4-х задач.

Виклад основного матеріалу. Зазначимо, що початкова мета розробки мови Паскаль диктувалася потребою у знаходженні інструмента «для навчання програмуванню як систематичній дисципліні». Pascal цікавий насамперед тим, що він став однією з перших реалізацій мов високого рівня написаних на самій собі, на два роки випередивши компілятор Сі. Pascal має жорстку типізацію даних. Це означає, що присвоювання можна виконувати лише для змінних, що належать до одного типу (одночасно вказуються правила, за якими типи вважаються однаковими). У такий спосіб суттєво покращується стиль програмування, оскільки значну частину помилок вдається виявити ще на етапі компіляції – це збільшує надійність програм. Відтак, переліковий тип задається переліком всіх значень, які може приймати змінна даного типу. Наприклад:

```
TYPE День = (понеділок, вівторок, середа, четвер, п'ятниця, субота, неділя);  
VAR Д: День;
```

Тут змінна Д типу День може приймати тільки вказані значення і ніяких інших. Тому доречно їх перелічити, доручивши таким чином системі програмування аналізувати змінну Д на наявність в ній тільки цих значень. Система програмування ставить у відповідність кожному значенню в переліку певний порядковий номер, починаючи з нуля, і працює саме з номером, а не з його текстовою відповідністю (в програмі текстові значення треба задавати транслітерацією). Порядок при переліченні має встановлені значення. У нашому прикладі вівторок<середа, вираз істинний, а середа>четвер – хибний.

Загальний формат опису змінної:

```
TYPE <назва типу> = (<значення 1>, ..., <значення n>);
VAR <назва типу>, або в розділі VAR - <назва змінної>:
(<значення 1>, ..., <значення n>).
```

Оскільки значення перелікового типу утворюють впорядковану множину, то для змінної такого типу існують функції **SUCC(x)**, **PRED(x)**, **ORD(x)**. Зі змінними перелікового типу можна виконувати всі оператори крім операторів вводу та виводу.

Наприклад:

```
TYPE
MISECH = (sichen, ljutyj, veresen, kviten,
traven, cherven, lipen, serpen, veresen, jovten,
listopad, gruden);
VAR M: MISECH;
... (Текст програми)
IF (M>traven) AND (M<veresen) THEN
  WRITELN('літній місяць');
  M: =veresen;
```

З допомогою оператора **WRITE(ORD(M))** буде виведено значення 2. Допускається використання змінної даного типу в ролі параметра циклу в операторі **FOR**. Наприклад:

```
FOR M: =sichen TO gruden DO
  BEGIN
    (Тіло циклу)
  END.
```

З погляду на вище вказане, пропонуємо на розсуд читача приклади таких задач:

***Задача 1.** В змаганнях приймають участь Валя, Гена, Коля, Маша, Ніна, Саша, Таня, Ваня, Шура. Відомий зріст кожного учасника. Визначити ім'я найвищого спортсмена серед хлопців та середній зріст дівчаток в групі спортсменів. Оскільки змінна учасник в програмі може приймати тільки вказані значення (Валя, Гена, Коля, Маша, Ніна, Саша, Таня, Ваня, Шура), то таку змінну доречно описати переліковним типом:*

```
TYPE imja=(valja, gena, enja, kolja, masha, nina, sasha, tanja, venja, shura) (див. програму*).
```

В програмі всі мовні конструкції будуть на значеннях змінної **I** цього типу. В результаті програма має такий вигляд:

***Приклад програми:**

```
program nom _1;
uses crt;
type imja=(valja,gena, jenja,kolja,masha,nina,sasha,tanja,venja,shura);
```

Розмір масиву задається діапазоном на переліковному типі

Масив задається константами перелікового типу і означає стать спортсмена (j -жіноча, m- чоловіча)

```
const stat :array [valja..shura] of (m,j)=(j,m,m,m,j,j,m,j,m,j );
      rost :array [imja] of 140..200=(150,170,180,190,145,150,189,164,170,187);
```

```
var I,I:imja;
```

```
sr,r:real;
```

```
k:integer;
```

```
begin
```

```
r:=rost[valja];k:=0; sr:=0;
```

```
for I:= valja to shura do
```

```
begin
```

```
if stat[I]=m then if r<rost[I] then begin r:=rost[I]; i1:= I end;
```

```
if stat [I]=j then begin sr:=sr+rost [I]; k:=k+1; end
```

```
end;
```

```
case i1 of
```

Змінна **i1** вміщує ім'я найвищого спортсмена серед хлопців

```
gena: write (' Гена');
```

```
jenja: write (' Женя');
```

```
kolja: write (' Коля');
```

```
sasha: write (' Саша');
```

```
venja: write (' Ваня');
```

```
shura: write (' Шура');
```

```
end;
```

```
write( ' - найвищий спортсмен його зріст ',r:4:0, ' ');
```

```
write( ' середній зріст спортсменок ',sr/k:4:0)
```

```
end.
```

Змінна перелікового типу може бути виведена на друк тільки програмно!

Жирним шрифтом виділені всі елементи програми, які пов'язані з переліковним типом, а також приведений загальноприйнятий фрагмент виводу на друк змінної перелікового типу. Курсивом відмічені коментарі, що не входять до складу програм.

Задача 2. В бюро погоди зберігається температура повітря за кожний день року. Визначити назву місяця з найвищою середньою температурою за рік.

Змінна, що означає місяць, може приймати тільки 12 визначених значень. Тому її доречно описати з допомогою перелікового типу, наприклад:

type misjac=(jen,feb,mar,apr,maj,jun,julaj,ougest,sep,nov,oct,dec) (див. програму**).

****Приклад програми:**

```

program nom_2;
uses crt;
type misjac =(jen,feb,mar,apr,maj,jun,julaj,ougest,sep,nov,oct,dec);
const kd:array[misjac] of 28..31=(30,28,31,30,31,30,31,30,31,30,31,30);
var m,m1:misjac;
t:array[1..365] of real ;
l,I,n:integer;
max,k:real;
begin clrscr;

```



```

for I:=1 to 365 do
t [I]:=random (30);

```

Випадковими числами задано температуру за кожний день

```

randomize;
max:=(t[1] ); m1:=jen; l:=0;
for m:=jen to dec do
begin

```

```

k:=0;
for n:=1 to kd[m] do
begin
k:=k+t [n];l:=l+kd[m];
if max < k then
begin max:=k;
m1:=m;
end;
end;

```



Цикл по місяцях організований за допомогою перелікового типу. Виконується 12 разів



```

jen :writeln ('sichni');
feb: writeln ('ljutomy');
mar : writeln ('berezni');
apr : writeln ('kvitni');
maj : writeln ('travni');
jun : writeln ('chervni');
julaj : writeln ('lipni');
ougest : writeln ('serpni');
oct : writeln ('jovtni');
nov : writeln ('listopadi');
dec : writeln ('grudni');

```

Виведення змінної m1

```

write('max serednja temperatura= ',max/kd [m1]:6:1, ' u ');
case m1 of
end;
end.

```



Програми, у котрих використовуються змінні перелікового типу набагато простіші для розуміння в своїй основній логічній структурі.

Задача 3. Дано середню температуру кожного місяця на кожному з 5 островів (Барбадос, Гаїті, Гренада, Куба, Мартініка, Джамайка). Визначити на якому острові та в якому місяці була найнижча середня температура.

Змінна, що вміщує назву острова має бути описана переліковим типом. Змінну, що вміщує назву місяця теж доречно описати переліковим типом. У програмі показано два варіанти реалізації алгоритму. Другий варіант відмічений курсивом (див. програму***).

*****Приклад програми:**

```

program nom_3;
uses crt;
type ostriv =(Barbados,Gaiti,Grenada,Kuba,Martinika,Jamajka);
   misjac =(jen,feb,mar,apr,maj,jun,julaj,ougest,sep,nov,oct,dec);
var t:array [ostriv,misjac] of real;
   o1,o:ostriv;
   m,m1:misjac;
I , n : integer ;
min : real ;
begin
  clrscr ;
  randomize ;
  for o := barbados to jamajka do
    begin
      for m :=jen to dec do
        t [ o , m ] :=random(30);
      end;
      (* m:=jen ;
        while m <dek do
          begin
            t [ o , m ] :=random(30);
            m:=succ (m)
          end;
        *)
min:=t[barbados , jen ] :
o:= barbados ; задається початкове значення
for t:=barbados to jamajka do
  begin
    for m:=jen to dec do
      begin
        if t [o;m]<min then
          begin min := t[ o ,m ];
            m1:=m
          end;
    end;
  end;

```

Обидва розміри двовимірного масиву описуються переліковим типом. Масив вміщує температури на відповідних островах.

Другий спосіб заповнення масиву

Перехід до наступного місяця

```

end;
end;
(* m:=jen ;
while m <dek
begin
  if min > t[ o ,m ] then
  begin
    min := t[ o ,m ]
    o1:=o; → Змінна o1 вміщує назву острова
    m1:=m → Змінна m1 вміщує назву місяця
  end;
  m:=succ (m)
end;
*)
write ('min serednja temperatura na ostrovi');
case o1 of
Barbados  :write (' Barbados ');
Gaiti      :write ('gaiti ');
Grenada    : write ('Grenada ');
Kuba       :write ('Kuba ');
Jjamajka   :write ('Jamajka ');
end;
write (' u ');
case m1 of

jen        :writeln ('sichni');
fed        :writeln (' Ijutomy');
mar        :writeln ('berezni');
apr        :writeln ('kvitni');
maj        :writeln ('travni');
jun        :writeln ('chervni');
julaj     :writeln ('lipni');
ougest    :writeln ('serpni');
oct        :writeln ('jovtni');
nov        :writeln ('listopadi');
dec        :writeln ('grudni');
end;
end.

```

Другий спосіб організації циклу

*Виведення назви острова
Martinika: write ('Martinika');*

Виведення назви місяця

Другий спосіб задання циклів в програмі демонструє більш наочну роботу зі змінними перелікового типу. В ньому демонструється послідовне заповнення змінних значеннями перелікового типу.

Задача 4. За введеним номером року нашої ери визначити його назву за японським календарем, за яким прийнято 60-річний цикл, що складається з 5 12-річних підциклів. Підцикли позначаються назвами зелений , червоний , жовтий , білий , чорний. У підциклі роки мали назви: курки, собаки, свині, криси,

корови, тигра, зайця, дракона, змії, коня, вівці, мавпи. 1984 рік – рік зеленої криси – був початком чергового циклу (див. програму ****).

******Приклад програми:**

```

program nom_4;
uses crt;
var rik : (kurka , sobaka , svinja , krisa , korova , tigr , zaec , drakon , zmija , kin ,
vivca , obizana);
podcikl : (zelenij , cervonij , zovtij , bilij , cornij);
I,n:integer;
begin
  clrscr ;
  writeln ('rik');
  read (n); I:=1;
  while n > 60 do
    n:=n-60;
    podcikl := zelenij;
    while n > 12 do
      begin
        n:=n-12; podcikl:=succ (podcikl)
      end;
    rik:=kurka;
    for I:=2 to n do
      rik:= succ (rik);

```

Змінна **n** вміщує остачу від ділення року на 60

Змінна **podcikl** отримує своє значення (наступне з перелічення при описі типу наочно і просто!)

Аналогічно отримує своє значення змінна **rik**

Задача розв'язана! Залишається тільки здійснити виведення значень на екран. Висновок об'ємний, але в основній логічній структурі використання перелікового типу має великі переваги:

```

case podcikl of
zelenij:
  begin
    write ('зелений');
    case rik of
      krisa: writeln (' криси');
      korova : writeln ('корова');
      tigr: writeln ('тигр');
      zaec: writeln ('заєць');
      drakon : writeln ('дракон');
      zmija: writeln ('змія');
      kin: writeln ('кінь');
      vivca: writeln ('вівця');
      obizana: writeln ('мавпа');
      kurka: writeln ('курка');
      sobaka: writeln ('собака');
      svinja: writeln ('свиня');

```

```
end;  
end;  
cervonij:  
begin  
write ('червоний');  
case rik of  
    krisa: writeln ('криса');  
    korova: writeln ('корова');  
    tigr: writeln ('тигр');  
    zaec: writeln ('заець');  
    drakon: writeln ('дракон');  
    zmija: writeln ('змія');  
    kin: writeln ('кінь');  
    vivca: writeln ('вівця');  
    obizana: writeln ('мавпа');  
    kurka: writeln ('курка');  
    sobaka: writeln ('собака');  
    svinja: writeln ('свиня');  
end;  
end;  
zovtij:  
begin write ('жовтий');  
case rik of  
    krisa: writeln ('криса');  
    korova: writeln ('корова');  
    tigr: writeln ('тигр');  
    zaec: writeln ('заець');  
    drakon: writeln ('дракон');  
    zmija: writeln ('змія');  
    kin: writeln ('кінь');  
    vivca: writeln ('вівця');  
    obizana: writeln ('мавпа');  
    kurka: writeln ('курка');  
    sobaka: writeln ('собака');  
    svinja: writeln ('свиня');  
end;  
end;  
bilij:  
begin write ('білий');  
case rik of  
    krisa: writeln ('криса');  
    korova: writeln ('корова');  
    tigr: writeln ('тигр');  
    zaec: writeln ('заець');  
    drakon: writeln ('дракон');
```

```
zmiya: writeln ('змія');  
kin: writeln ('кінь');  
vivca: writeln ('вівця');  
obizana: writeln ('мавпа');  
kurka: writeln ('курка');  
sobaka: writeln ('собака');  
svinja: writeln ('свиня');  
end;  
end;  
end;  
end.
```

Висновок. Приведені задачі можна розв'язати і не використовуючи перелікового типу, але такий розв'язок був би набагато складнішим, мало зрозумілим і неефективним. Існує цілий клас задач, подібних до приведених, при розв'язуванні яких переліковий тип даних є найбільш правильний результат.

Список використаних джерел

1. Бородич Ю.С., Вальвачев А.Н., Кузьмич А.И. Паскаль для персональных компьютеров / Ю.С. Бородич, А.Н. Вальвачев, А.И. Кузьмич. – Минск, 1991. – 367 с.
2. Глосарій термінів STEM-освіти. – [Електронний ресурс]. – Режим доступу до ресурсу: http://ontology.inhost.com.ua/index.php?graph_uid=1347).
3. Єльнікова Г.В. STEM-освіта в контексті адаптивного підходу [Електронний ресурс] / Г.В. Єльнікова // Адаптивне управління: теорія і практика. Серія «Педагогіка». – 2018. – № 4 (7). – Режим доступу до ресурсу : http://am.eor.by/images/adapt/Vol.4ped7/18ped4_7yelnikova.pdf.
4. Караванова Т.П. Інформатика. Основи алгоритмізації та програмування (процедурне програмування) : [навч. посібник] / Т.П. Караванова. – Шепетівка : Аспект, 2005. – 250 с.
5. Культин Н.Б. Delphi 6. Программирование на Object Pascal / Н.Б. Культин.– М., 2002. – 526 с.

6. Наукові записки Малої академії наук України. Серія «Педагогічні науки» : [зб. наук. праць ; редкол. : С.О. Довгий (голова), О.Є. Стрижак, О.В. Лісовий, І.М. Савченко та ін.]. – К. : Інститут обдарованої дитини НАПН України, 2018. – Вип. 9. – 210 с.

7. Наукові записки Малої академії наук України. Серія «Педагогічні науки» : [зб. наук. праць ; редкол. : С.О. Довгий (голова), О.Є. Стрижак, О.В. Лісовий, І.М. Савченко та ін.]. – К. : Інститут обдарованої дитини НАПН України, 2018. – Вип. 10. – 226 с.

8. Пильщиков В.Н. Сборник упражнений по языку Паскаль / В.Н. Пильщиков. – М. : Вид-во «Наука», 1989. – 1600 с.

9. Програмування : [практикум] / укл. : А.Д. Семенюк, Ф.О. Сопронюк. – Чернівці : Рута, 2001. – 143 с.

10. Розбудова єдиного інформаційного простору української освіти – вимога часу : [зб. матер. Всеукр. наук.-практ. WEB-форуму (Київ-Харків, 22–23 березня 2018 р.); за заг. ред. : М.Л. Ростока, І.М. Савченко, Т.С. Бондаренко]. – Кропивницький : Вид-во Льотної академії Національного авіаційного університету, 2018. – 184 с.

11. STEM-освіта – проблеми та перспективи: [зб. матер. II Міжнар. наук.-практ. семінару (м. Кропивницький, 256 жовтня 2017 р.)] / за заг. ред. О.С. Кузьменко та В.В. Фоменка. – Кропивницький : КЛА НАУ, 2017. – 120 с.

12. STEM-освіта: стан впровадження та перспективи розвитку : [матер. III Міжнар. наук.-практ. конф. (м. Київ, 9–10 листопада, 2017 р.)]. – ДНУ «Інститут модернізації змісту освіти МОН України», 2017. – 160 с.

13. Фаронов В.В. Delphi. Программирование на языке высокого уровня / В.В. Фаронов. – СПб. : Питер, 2004. – 640 с.

14. Юрченко І.В. Інформатика та програмування : [навч. посібник] / І.В. Юрченко. – Чернівці : Книги–XXI, 2011. – Ч. 1. – 203 с.

15. Юрченко І.В., Сікора В.С. Інформатика та програмування / І.В. Юрченко, В.С. Сікора. – Ч. 2.– Чернівці : Вид-ць С.Н. Яворський, 2015. – 210 с.

Косинова Надежда. ИСПОЛЬЗОВАНИЕ ПЕРЕЧИСЛЯЕМОГО ТИПА ПРИ РЕШЕНИИ ЗАДАЧ НА ЯЗЫКЕ ПАСКАЛЬ

Предлагается набор задач, решенных на языке Паскаль, в которых использованы данные перечисляемого типа. Использование данных перечисляемого типа упрощает решение целого класса типовых задач при изучении программирования. Каждый язык программирования имеет средства для решения таких задач.

В основе лежит принцип «Лучше один раз увидеть чем сто раз услышать». Приведены готовые программы, в которых сделаны акценты на использование перечисляемого типа и приведены теоретические выкладки о перечисляемом типе в языке Pascal.

Рассчитана для лиц, изучающих язык Паскаль или преподающих его.

Ключевые слова: *STEM-специальность, перечисляемый тип, язык программирования, программа, данные, массив данных.*

Kosinova Nadia. USE OF THE ENUMERATED TYPE WHEN SOLVING PROBLEMS IN THE LANGUAGE PASCAL

A set of problems solved in the Pascal language is proposed, in which the data of the enumerated type are used.

Using data of the enumerated type simplifies the solution of a whole class of typical problems in the study of programming. Each programming language has the means to solve such problems.

The basis is the principle "It is better to see once than hear a hundred times." Ready-made programs are given, in which emphasis is placed on the use of the enumerated type and theoretical calculations are given about the enumerated type in the language Pascal.

Designed for individuals who study Pascal or teach it.

Key words: *STEM-specialty, enumerated type, programming language, program, data, data array.*