Тема: Алгоритми і програми з розгалуженнями.

План.

1. Оператор безумовного переходу (goto).
2. Умовні оператори.
3. Складена команда.

Література: Я.М. Глинський. Інформатика. Алгоритмізація і програмування. Книжка 1. с. 67-85.

1 Оператор безумовного переходу (goto).

Команду переходу використовують щоб змінити порядок виконання команд у програмі.

goto <міткa>

Вона має такий вигляд:

goto – перекладається “перейти на”, де “мітка” – це адрес команди (оператора), на який потрібно перейти. Мітка може бути числом від 0 до 9999 або ідентифікатор, після якого є символ (:).

 Приклади міток: 10:, 222:, m1:, alpha:. Усі мітки можна оголосити у розділі оголошення міток на початку програми так:

label <список міток>

Наприклад:

 label 10:, 222:, m1:, alpha:.

**Задача** Скласти програму обчислення функції:



 для трьох різних значень x = 3,4,5 з клавіатури з використанням команди переходу goto.

Програма:

program FUN\_Y;

 label 10;

 var x,y: real;

begin

 10: write (’введіть значення х:’);

 read ln (x);

 y: = (3\*5 sin (x) + cos (2\*x)/(3.5-4\*abs(x));

 write ln (x:6:2, y:6:2)

 goto 10;

end.

2. Умовні оператори.

 У повсякденній діяльності часто доводиться аналізувати різноманітні ситуації, умови і залежно від них приймати ті чи інші рішення.

 Команди розгалуження або умовні оператори змінюють порядок виконання дій у алгоритмі в залежності від виконання чи невиконання певних умов. Розгалуження бувають декількох видів. Найпростішим видом є “неповне розгалуження”.

Неповне розгалуження реалізується за допомогою команди if (якщо):

if <умова> then <команда>

Виконання команди: <команда> може бути один оператор або група операторів. Якщо виконується умова (в блок-схемі – вітка “так”), то виконається оператори чи група операторів після слова then. Якщо ж умова не виконується (в блок-схемі – вітка “ні”), то ця команда не буде виконуватися і буде виконуватися наступний після if оператор (команда).

 Умова – це логічний вираз, які бувають прості і складені.

 Простий логічний вираз – це два арифметичні вирази, з’єднані символом відношення (порівняння).

 У Паскалі символи відношень записуються так:>-більше,<менше,

>= - більше або дорівнює (не менше), <= - менше або дорівнює (не більше),

= - дорівнює, < > - не дорівнює.

 Наприклад: x<5; x+y>=2\*x; x< >y.

 Складені логічні вирази – це прості логічні вирази, до яких застосовані логічні операції not (не), and (і), or (або).

 Наприклад. Побудувати вираз, який буде використовуватися для визначення студентів, які не отримали оцінки “2” з інформатики (тобто отримали інші оцінки). Для позначення оцінки використаємо числову змінну ocinka. Шуканий вираз можна записати так:

 А). not(ocinka=z) або ocinka< >2.

 Приклад.

 Записати подвійну нерівність

0<x<3 для дійсних х можна так: (x<0) and (x<3).

 Примітка. Прості логічні вирази, які входять у складені, записують у круглих дужках.

 Приклад.



 На мові Паскаль таку складену умову можна записати так:

x< >4 or x< >-5.

 Задача. Скласти програму, де у змінні а та b ввести два цілих числа. Третій змінній с присвоїти значення більшого з-поміж уведених. Якщо числа рівні, то змінні с присвоїти значення будь-якого з чисел.

program twonumbers;

var a,b, c: integer;

begin

 write (’Введіть два числа’);

 read ln(a,b);

 if a>b then c:=a;

 if a<b then c:=b;

 if a=b then c:=a;

 write ln (‘c=’, c); read ln

end.

Задача. Увести будь-яке значення х і обчислити значення функції у, яка задана формулою



program F1

var x,y: real;

begin

 write (‘Введіть х:’);

 read ln (x);

 if x>0 then y:=x+3 else y:=x-5;

 write ln (‘y=’,y);

 read ln

end.

Повне розгалуження.

 Повне розгалуження реалізують за допомогою повної команди розгалуження if.

If<умова>then<команда 1>else<команда 2>;

 Виконання команди:

 Якщо умова виконується (в блок-схемі – вітка “так”), то виконується команда 1 (або серія команд 1), якщо ні – то команда 2 (серія команд 2).

 Зауваження. На місці команди 1 та команди 2 може бути теж команда розгалуження if.

 Приклад. Визначимо більше число з-поміж двох чисел:

 If a>=b then c:=a else c:=b;

де a i b – два числа, а с – проміжна змінна куди заносимо більше з 2-х чисел.

 Оператор вибору case.

 Якщо потрібно здійснити вибір з великої кількості варіантів, зручно використовувати оператор (команду) вибору case:

case <вираз-селектор>of

<список 1>:<команда 1>;

<список 2>:<команда 2>;

…

<список n>:<команда n>;

else <команда>

end;

 Виконання оператора:

 Якщо значення виразу збігається зі значенням з найбільшого списку чи діапазону, то виконується відповідна команда, що записана після двокрапки “:” і на цьому виконання команди case закінчується, якщо ні, то виконується команда після слова else. Складової частини else <команди> може бути.

 Задача. У багатьох університетах поточні знання студентів протягом семестру оцінюють за стобальною системою. Потім бали переводять в оцінки: <<відмінно>>,<<добре>>,<<задовільно>>,<<незадовільно>>.

 Оцінки визначають так:

 Бали Оцінка

 88-100 <<відмінно>>

 71-87 <<добре>>

 50-70 <<задовільно>>

 0-49 <<незадовільно>>

 Скласти програму, яка для конкретного студента (name) і його кількості балів (bal) виводить його оцінку (mark).

program Marks;

 var bal: integer; name, mark: string;

begin

 write (‘Введіть прізвище’);

 readln (name);

 write (‘Введіть бали’);

 readln (bal);

case bal of

 88…100: mark:= ‘відмінно’;

 71…87: mark:= ‘добре’;

 50…70: mark:= ‘задовільно’;

 0…49: mark:= ‘незадовільно’;

 else mark: = ‘0’

end;

if mark < > ‘0’ then writeln (name:15, mark:15)

else writeln (‘помилка під час введення кількості балів’):

end.

або інший приклад:

 case k of

 1; cina:=22,

 2,3: cina:=35;

 4: cina:=50;

 else cina:=97

 end;

3. Складена команда.

 Складена команда – це команда, в якій декілька команд об’єднано в одну за допомогою службових слів begin та end:

 begin

 <команда 1>;

 <команда 2>;

 …

 <команда n>;

 end;

Задача. Скласти програму, яка дає довідку про назву столиці (St) та кількість населення (nas, у мільйонах) деякої країни (kr) з такого переліку: Угорщина, Італія, Україна.

 program Countries;

 var kr, st: string; nas: integer;

 begin

 write (‘Введіть назву країни’);

 readln (kr);

 if kr = ‘Угорщина’ then

 begin

 st:= ‘Будапешт’;

 nas:=11

 end;

 if kr = ‘Італія’ then

 begin

 st:= ‘Рим’;

 nas:=60

 end;

 if kr = ‘Україна’ then

 began

 st:= ‘Київ’;

 nas:= 48

 end;

 writeln (‘Столиця - ’, st, ‘населення-’, nas, ‘млн осіб’)

 end.