Лабораторна робота №1
(Програмування І 2017/18 М.М. Древаль)

Використання основних керуючих конструкцій

Мета роботи: Отримання навичок роботи з основними керуючими конструкціями С.

1.1. Теоретичні відомості

В мові програмування С використовуються наступні керуючі конструкції:

1) Цикл `for`

```
for (вираз1; вираз2; вираз3)
   {оператор}
```

Вираз1 описує ініціалізацію циклу; *вираз2* — перевіряє умову завершення цикла, якщо він істинний то виконується *оператор*; *вираз3* — обчислюється після кожної ітерації. Цикл повторюється до тих пір, поки *вираз* не стане хибним.

Наприклад:

```
for(x=1; x<=7; x++)
  printf("%d\n", power(x, 2));
```

2) Цикл `while`:

```
while (вираз)
  оператор
```

Якщо *вираз* є істинним, то *оператор* виконується до тих пір, поки *вираз* не стане хибним. Якщо *вираз* хибний з самого початку, то керування передається наступному оператору. При цьому цикл не виконується зовсім. Значення *виразу* визначається до виконання оператора.

Наприклад:

```
while (k<n) {
  y=y*x; k++;
}
```

3) Цикл `do-while`. Його синтаксис:

```
do
  оператор
while(вираз);
```

Якщо `вираз` істинний, то `оператор` виконується та знову обчислюється значення `виразу`. Це повторюється, поки `вираз` не стане хибним. `Оператор` виконується не менше одного разу.

Наприклад:

```
x=1;
do
  printf("%d\n", power(x, 2));
while(++x<=7);
```

Оператор `do-while` перевіряє умову після виконання операторів циклу.

Оператор `while` перевіряє умову перед виконанням операторів циклу.

4) Перемикач `switch` має синтаксис:

```
switch (вираз) {
   case константа1: оператор
   case константа2: оператор
   case константа3: оператор
    ...
   default: оператор
}
```

Порівнює значення `виразу` з константами у всіх варіантах case та передає керування оператору, що відповідає значенню `виразу`. Якщо жодна з констант не відповідає значенню `виразу`, тоді керування передається оператору з міткою `default`. Якщо мітка `default` відсутня, не виконується ніяких дій.

Наприклад:

```
switch(x)
  case 2:
  case: y=3;
```

5) Оператор `break` перериває виконання найближчого вкладеного зовнішнього оператора `switch`, `while`, `do` або `for`. Керування передається наступному оператору.

Наприклад:

```
for(i=0; i<n; i++)
  if((a[i] = b[i])==0)
    break;
```

6) Оператор продовження `continue` передає керування на початок найближчого зовнішнього оператора циклу `switch`, `while`, `do` або `for`, та викликає початок наступної ітерації.

7) Оператор повернення `return`. Перериває виконання поточної функції та повертає керування програмі, що її викликала.

1.2. Приклад

Знайти усі прості нескоротні дроби між 0 та 1, знаменники яких не перевищують 7 (дріб задається двома натуральними числами — чисельником та знаменником).

```
#include <stdio.h>
#include <math.h>

int main()
{
     int i, j, l, k, i1, j1, k2;
     printf("Початок програми\n\n");
     /*Задання циклів*/
     for(i1=2;i1<8;++i1)
          for(j1=1;j1<i1;++j1)
          {
               /*Виведення дробу вигляду 1/n*/
               if(j1==1) printf("%d%d\n", j1,i1);
               l=0; i=i1; j=j1;
               /*Цикл для визначення НСД*/
               do
               {
                    k=i-j;
                    k2=i1%j1;
                    if(((k==0)&&(j!=1)) || (k2==0)) l=1;
                    i=j; j=k;
               } while(k>0);
               /*Виведення дробу, якщо флаг не піднято*/
               if(l!=1) printf("%d%d\n", j1, i1);
          }
          return 0;
}
```

1.3. Порядок виконання роботи

1. Проаналізувати умову задачі.
2. Створити алгоритм розв’язання задачі згідно з номером варіанту, оформити його у графічному вигляді (у вигляді блок-схеми загального алгоритму).
3. Розробити програмний продукт.
4. Протестувати програмний продукт на контрольних прикладах, зробити висновки стосовно подальших можливостей його удосконалення.
5. Результати роботи оформити протоколом.

1.4. Варіанти завдань

1-1. Дано натуральне число n. Знайти n перших простих чисел.

1-2. Задані натуральні числа m та n. Знайти натуральне число у проміжку від m до n з найбільшою сумою дільників. Перебір варіантів вивести на екран.

1-3. Дано натуральне число n. Знайти всі прості дільники цього числа.

1-4. Знайти всі досконалі числа, менші за задане натуральне число. Число називається досконалим, якщо воно дорівнює сумі своїх дільників (окрім самого себе).

1-5. Задані натуральні числа m та n. Знайти всі числа, менші за n, сума цифр яких дорівнює m.

1-6. Дано натуральне число n ($n \geq 2$). Знайти всі прості числа, що не перевищують число n, використовуючи сито Ератосфена. Ситом Ератосфена називається наступний метод. Випишемо підряд всі цілі числа від 2 до n. Першим простим чилом є число 2. Підкреслимо його, а всі більші за нього числа, кратні числу 2, викреслимо. Першим числом в ряду чисел, що залишились, є число 3. Підкреслимо його як просте число, а всі більші за нього числа, кратні числу 3, викреслимо. Першим числом серед тих, що залишились, є число 5, оскільки 4 вже закреслено. Підкреслимо його як просте число, а всі більші за нього числа, кратні числу 5, викреслимо і т.д.

1-7. Дано натуральне число n. Знайти всі числа Мерсена, що не перевищують число n. Просте число називається *числом Мерсена*, якщо його можна представити у вигляді $2^p - 1$, де p – теж просте число.

1-8. Два натуральних числа називаються дружніми, якщо кожне з них дорівнює сумі всіх дільників іншого, крім самого цього числа. Знайти всі пари дружніх чисел, що містяться у діапазоні від m до n (значення m та n задаються користувачем).

1-9. Дано натуральне число n. Отримати всі натуральні числа q, такі, що n ділиться на q^2 та не ділиться на q^3.

1-10. Дано натуральне число n. Серед чисел 1, ..., n знайти такі числа, запис яких співпадає з останніми цифрами запису їх квадрату. Наприклад, 6 ($6^2 = 36$), 25 ($25^2 = 625$) і т.д.

1-11. Дано натуральне число n. Натуральне число називається паліндромом, якщо його запис читається однаково зліва направо та справа наліво (наприклад, 4884, 393, 1). Знайти всі числа-паліндроми, що не перевищують n, та які при піднесенні до квадрату також дають паліндроми.

1-12. Дано натуральне число n. Перевірити, чи можна подати $n!$ у вигляді добутку трьох послідовних цілих чисел.

1-13. Дано матрицю розміром n на m з дійсними елементами. Побудувати послідовність $b_1, b_2, \ldots, b_n$, де b_k – це кількість від'ємних елементів k-го рядка матриці.

1-14. Дано матрицю розміром n на m з дійсними елементами. Побудувати послідовність $b_1, b_2, \ldots, b_n$, де b_k — це добуток квадратів тих елементів k-го рядка, модулі яких належать відрізку [1, 1.5].

1-15. Дано натуральне число n та цілочисельна матриця порядку n. Побудувати послідовність $b_1, b_2, \ldots, b_n$, де b_k — це найменше із значень елементів, що розташовані на початку k-го рядка матриці (до головної діагоналі, включаючи елемент, що належить головній діагоналі).

1-16. Дано натуральне число n та цілочисельна квадратна матриця порядку n. Побудувати послідовність $b_1, b_2, \ldots, b_n$, де b_k — це сума елементів, що передують останньому від'ємному елементу k-го рядка матриці (якщо всі елементи рядка невід'ємні, то вважати $b_k = -1$).

1-17. Дано натуральне число n. Натуральне число, що складається з n цифр, є *числом Армстронга*, якщо сума його цифр, піднесених до n-го ступеня, дорівнює самому числу (наприклад, $153 = 1^3 + 5^3 + 3^3$). Знайти всі числа Армстронга, що складаються не більше ніж з n цифр.

1-18. Дано натуральні числа $n, b_1, b_2, \ldots, b_n$. Знайти члени b_k послідовності $b_1, b_2, \ldots, b_n$, що мають таку властивість: корені рівняння $x^2 + 3b_k - 5$ є дійсними та додатніми.

1-19. Дано натуральне число n та дійсні числа $b_1, b_2, \ldots, b_n$. Обчислити зворотну величину для добутку тих чисел b_k послідовності $b_1, b_2, \ldots, b_n$, для яких виконується умова $k + 1 < b_k < k!$.

1-20. Дано натуральне число n. Обчислити суму тих чисел вигляду $i^3 - 3in^2 + n$, $(i = 1, 2, \ldots, n)$, що являють собою потроєні непарні числа.

1-21. Дано цілочисельну квадратну матрицю порядку n. Знайти номери тих рядків, елементи котрих утворюють симетричні послідовності (паліндроми).

1-22. Дано натуральні числа $n, b_1, b_2, \ldots, b_n$. Знайти ті члени b_k послідовності $b_1, b_2, \ldots, b_n$, що при діленні на 7 дають остачу 1, 2 або 5.

1-23. Дано натуральні числа $n, b_1, b_2, \ldots, b_n$. Знайти члени b_k послідовності $b_1, b_2, \ldots, b_n$, що є потроєними непарними числами.

1-24. Дано натуральні числа $n, b_1, b_2, \ldots, b_n$. Знайти кількість членів b_k послідовності $b_1, b_2, \ldots, b_n$, що мають парні порядкові номери, та є непарними числами.

1-25. Дано натуральні числа $n, b_1, b_2, \ldots, b_n$. Знайти члени b_k послідовності $b_1, b_2, \ldots, b_n$, що є подвоєними непарними числами.

1-26. Дано квадратну матрицю порядку n з дійсними елементами. Обчислити суму тих її елементів, що розташовані на головній діагоналі та вище від неї, та не перевищують по величині суму елементів, розташованих нижче головної діагоналі. Якщо на головній діагоналі та вище від неї елементів з такою властивістю немає, то відповіддю повинно бути відповідне повідомлення.

1-27. Дано квадратну матрицю порядку n з цілими елементами. Знайти найменше із значень елементів стовбчика, сума модулів елементів котрого є найбільшою. Якщо таких стовбчиків існує декілька, то взяти перший із них.

1-28. Дано натуральне число n і послідовність b_j ($j = 1{,}2n$) дійсних чисел така, де кожні 2 сусідні елементи – це координати x і y відповідної точки. Надрукувати номери двох найбільш віддалених одна від одної сусідніх точок.

1-29. Дано $2n$ дійсних чисел b_j. Відомо, що кожні 2 сусідні числа – декартові координати відповідної точки. Знайти площу, обмежену n-кутником з вершинами у відповідних точках, якщо відомо, що даний n-кутник є опуклим і перша точка з'єднана з n-ю.

1-30. Дано натуральні числа n, b_1, ..., b_n. Кожне з b_j – це порядковий номер дня високосного року. Надрукувати всі номери субот та неділь, що потрапляють до цієї послідовності, якщо відомо, що 1 січня було вказаним користувачем днем тижня (надати користувачу можливість задати день тижня шляхом вибору зі списку).

1-31. Задано n точок у полярній системі координат так, що 2 чергових елемента послідовності b_j і b_{j+1} є R_j і φ_j . Надрукувати декартові координати всіх точок послідовності, що потрапили до вказаної користувачем чверті.

1-32. Задано n цілих чисел. Вивести на екран всі числа послідовності, у десятковому записі яких присутні тільки парні цифри.

1-33. Задано n цілих чисел. Вивести на екран всі числа послідовності, у двійковому записі яких присутні тільки одиниці.

1.5. Контрольні запитання

1. Що таке проста та складна інструкції?
2. Цикл з передумовою, цикл з постумовою. Їх синтаксис.
3. Що таке покроковий цикл?
4. Коли використовується перемикач?
5. Призначення конструкцій продовження, безумовного переходу, повернення та переривання у циклах різного роду.