Курс «Основы программирования»

Власенко Олег Федосович

SimbirSoft

Лекция 6

Управление:

IF, GOTO, DO WHILE, WHILE, RETURN, CONTINUE, BREAK, SWITCH

ЛР 10. Использование SWITCH

ЛР 11. Цикл WHILE

Где прочитать про управление в Си?

http://givi.olnd.ru/kr2/index.html - *Б. Керниган, Д. Ритчи. Язык программирования Си*

http://givi.olnd.ru/kr2/03.html - *Глава 3. Управление*

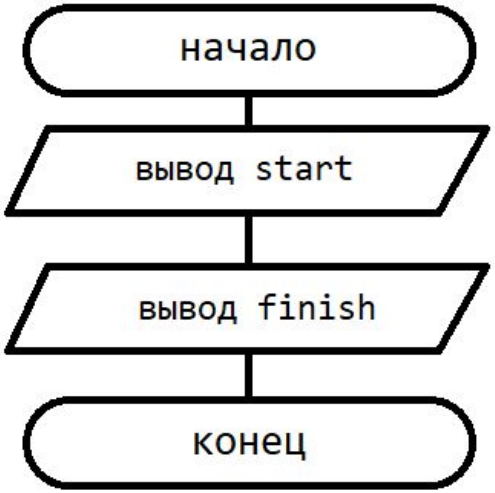
Глава 3. Управление

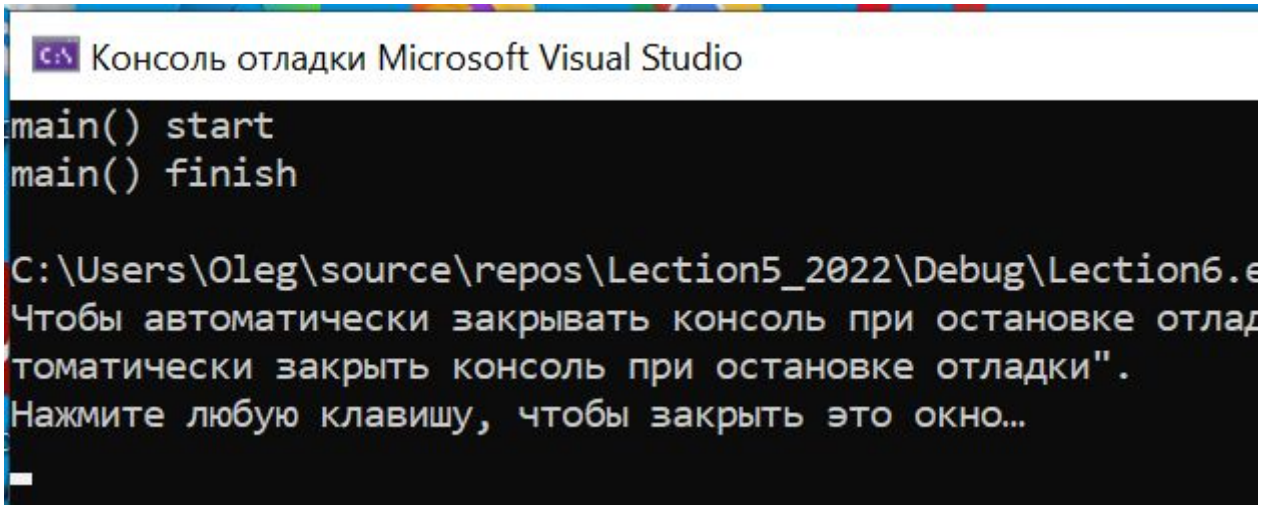
- 3.1. Инструкции и блоки
- 3.2. Конструкция if-else
- 3.3. Конструкция else-if
- 3.4. Переключатель
- 3.5. Циклы while и for
- 3.6. Цикл do-while
- 3.7. Инструкции break и continue
- 3.8. Инструкция goto и метки

Простейшая программа (естественно, на Си)

Функция main() - точка входа в программу.

```
#include <stdio.h>

void main() {
    printf("main() start\n");

    printf("main() finish\n");
}
```

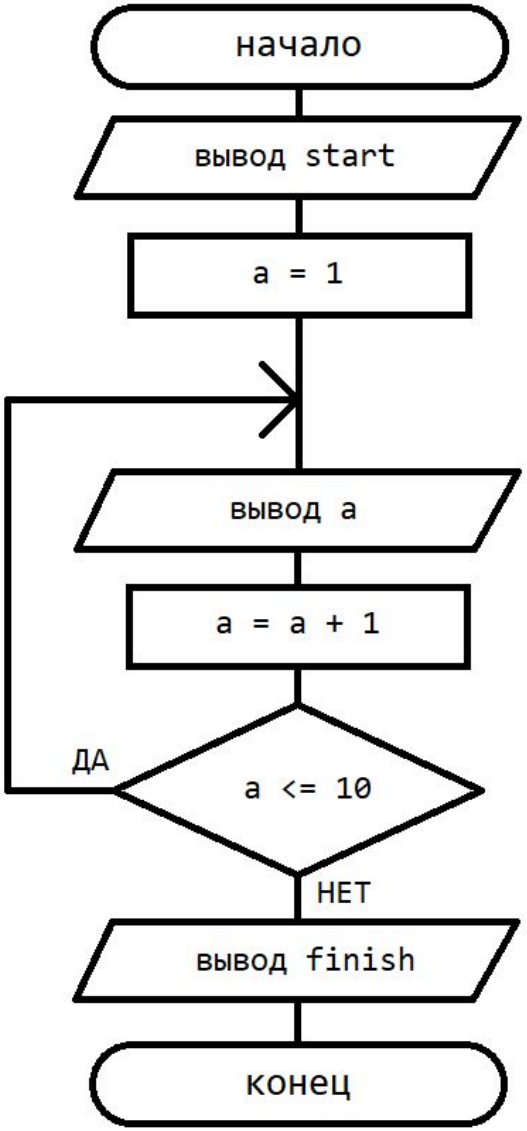


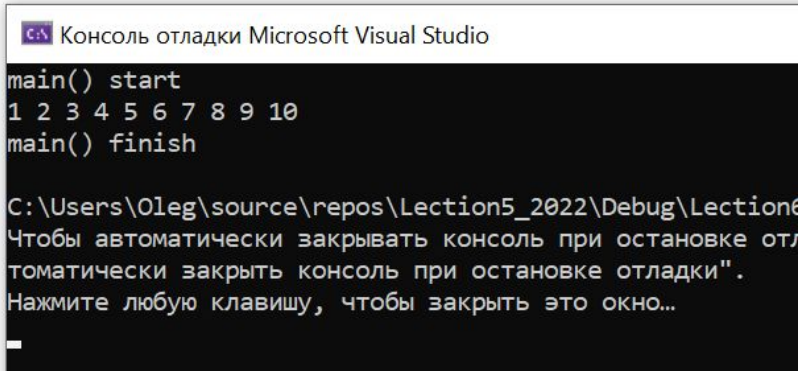


Простейший цикл DO WHILE

Вывод чисел от 1 до 10 - через DO WHILE

```
#include <stdio.h>
void main() {
    printf("main() start\n");
    int a = 1;
    do {
            printf("%d ", a);
            a = a + 1;
    } while (a <= 10);
    printf("\nmain() finish\n");
}
```

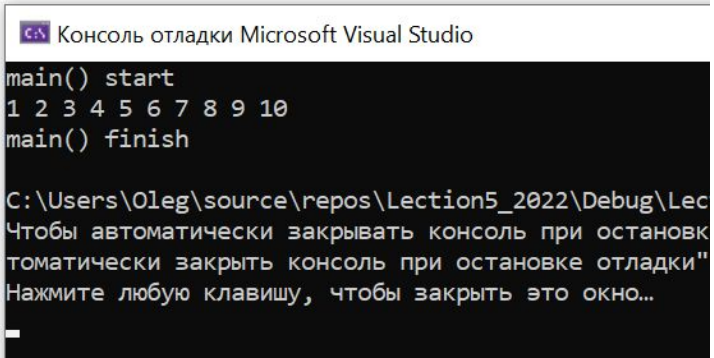


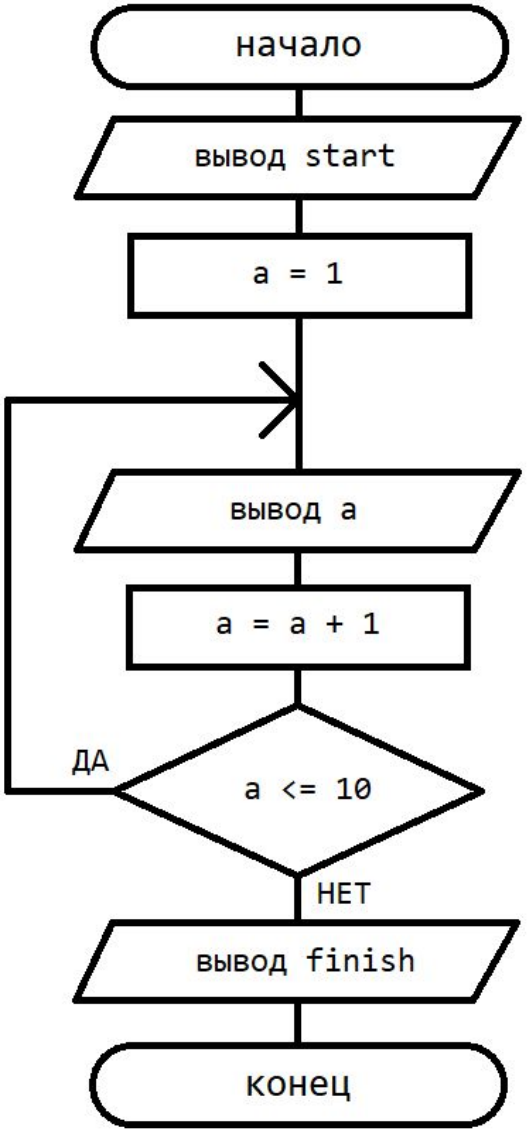


Знакомство с GOTO

Вывод чисел от 1 до 10 - через GOTO (эмуляция DO WHIL

```
#include <stdio.h>
void main() {
    printf("main() start\n");
    int a = 1;
    //do {
label_do:
    printf("%d ", a);
    a = a + 1;
    //} while (a <= 10);
    if (a <= 10) goto label_do;
    printf("\nmain() finish\n");
}
```

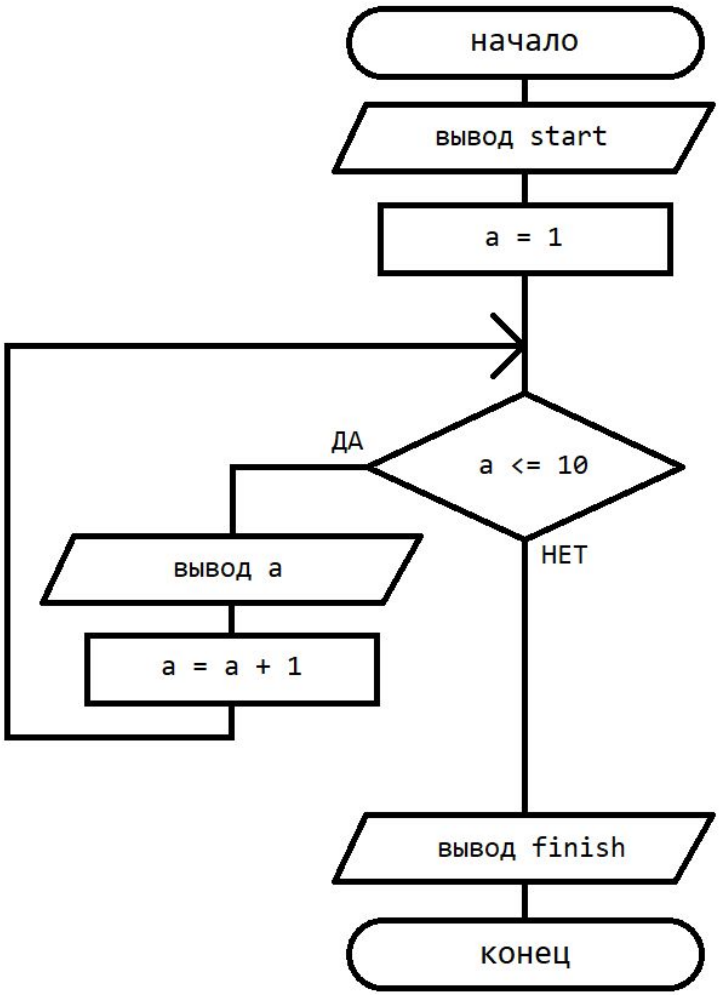


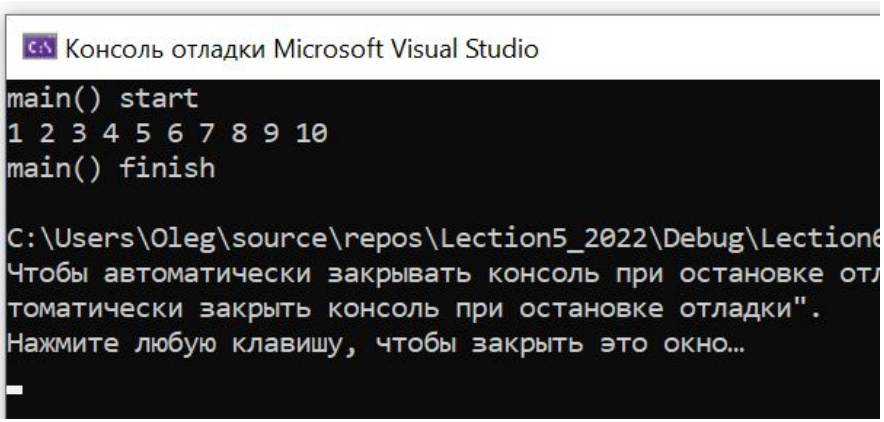


Простейший цикл WHILE

Вывод чисел от 1 до 10 - через WHILE

```
#include <stdio.h>
void main() {
    printf("main() start\n");
    int a = 1;
    while (a <= 10) {
        printf("%d ", a);
        a = a + 1;
    }
    printf("\nmain() finish\n");
}
```

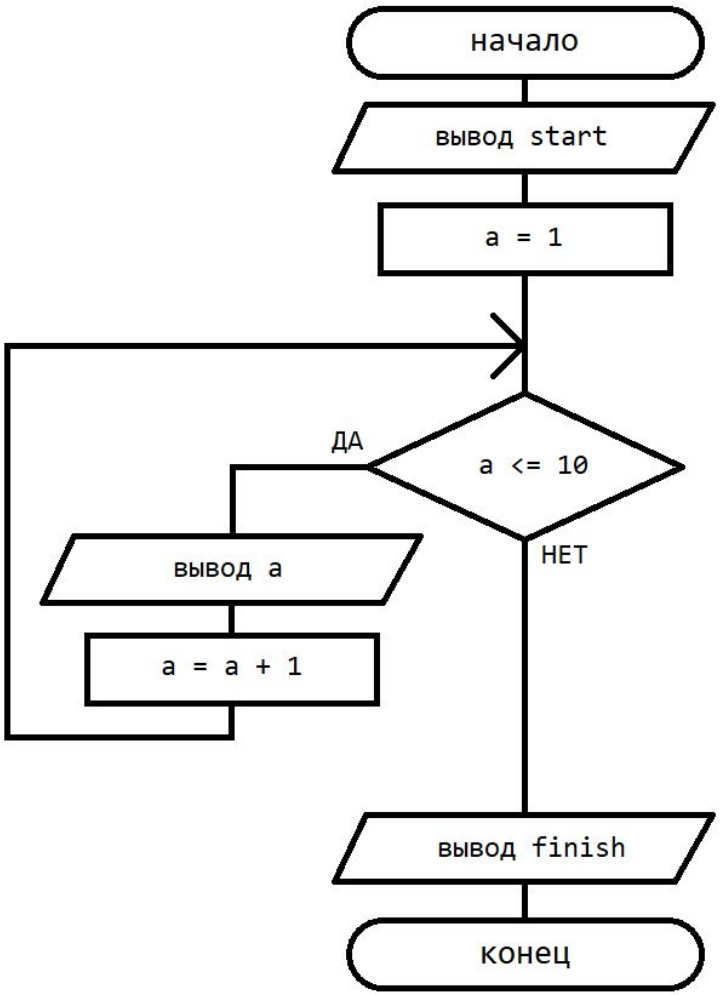




GOTO vs WHILE

Вывод от 1 до 10 - через GOTO (эмуляция WHIL

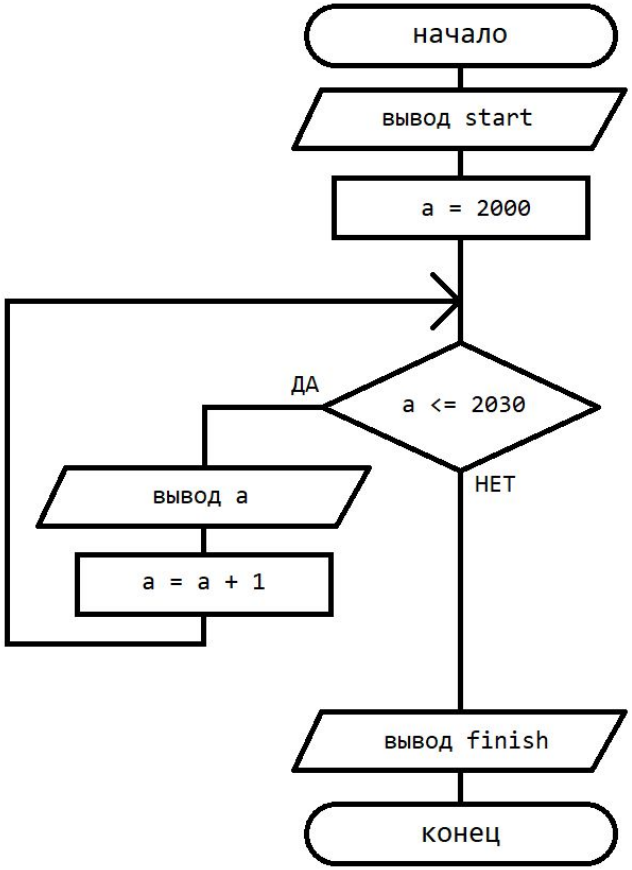
```
#include <stdio.h>
void main() {
    printf("main() start\n");

    int a = 1;

    // while (a <= 10) {
label_while:
    if (a <= 10) {
        printf("%d ", a);
        a = a + 1;
        goto label_while;
    }

    printf("\nmain() finish\n");
}
```



WHILE

Вывод 2000 2001 2002 2003 … 2030 - через WHILE

```
#include <stdio.h>
void main() {
    printf("main() start\n");

    int a = 2000;
    while (a <= 2030) {
        printf("%d ", a);
        a = a + 1;
    }

    printf("\nmain() finish\n");
}
```



Консоль отладки Microsoft Visual Studio

```
main() start
2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023
2024 2025 2026 2027 2028 2029 2030
main() finish

C:\Users\Oleg\source\repos\Lection5_2022\Debug\Lection6.exe (процесс 11560) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Ав
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно…
```

WHILE vs WHILE

Вывод 2000 2001 2002 2003 … 2030 - через WHILE – двумя разными способами

```c
#include <stdio.h>
void main() {
    printf("main() start\n");

    int a = 2000;
    while (a <= 2030) {
        printf("%d ", a);
        a = a + 1;
    }

    printf("\nmain() finish\n");
}
```

```c
#include <stdio.h>
void main() {
    printf("main() start\n");

    int a = 1999;
    while (a < 2030) {
        a = a + 1;
        printf("%d ", a);
    }

    printf("\nmain() finish\n");
}
```

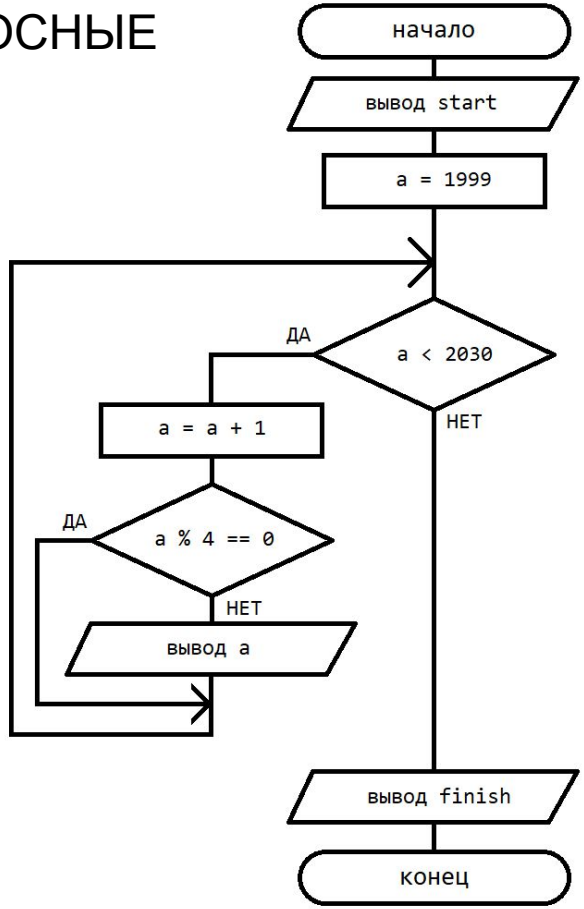
Консоль отладки Microsoft Visual Studio

```
main() start
2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023
2024 2025 2026 2027 2028 2029 2030
main() finish

C:\Users\Oleg\source\repos\Lection5_2022\Debug\Lection6.exe (процесс 11560) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Ав
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно…
```

CONTINUE

Вывод 2000 2001 2002 2003 … 2030 - только НЕВИСОКОСНЫЕ

```
#include <stdio.h>
void main() {
    printf("main() start\n");

    int a = 1999;
    while (a < 2030) {
        a = a + 1;
        if (a % 4 == 0)
            continue;
        printf("%d ", a);
    }

    printf("\nmain() finish\n");
}
```



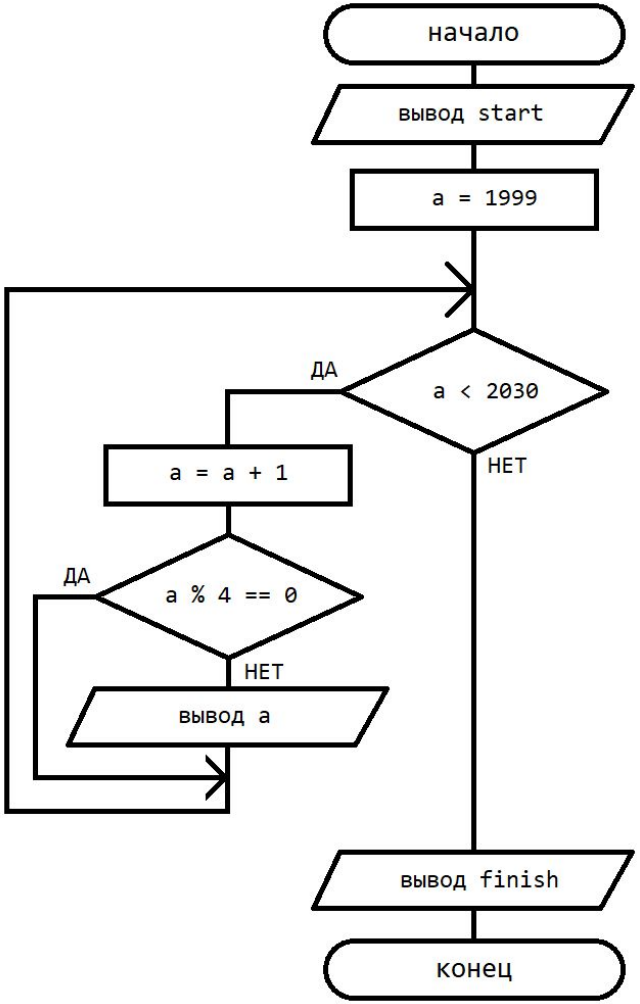
Консоль отладки Microsoft Visual Studio

```
main() start
2001 2002 2003 2005 2006 2007 2009 2010 2011 2013 2014 2015 2017 2018 2019 2021 2022 2023 2025 2026 2027 2029 2030
main() finish

C:\Users\Oleg\source\repos\Lection5_2022\Debug\Lection6.exe (процесс 18192) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Ав
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно…
```

WHILE & CONTINUE vs GOTO

Вывод 2000 2001 2002 2003 ... 2030 - только НЕВИСОКОСНЫЕ

```
#include <stdio.h>

void main() {
    printf("main() start\n");

    int a = 1999;
    while (a < 2030) {
        a = a + 1;
        if (a % 4 == 0)
            goto label_body_end;
        printf("%d ", a);
    label_body_end:;
    }

    printf("\nmain() finish\n");
}
```



WHILE & CONTINUE vs GOTO (2)

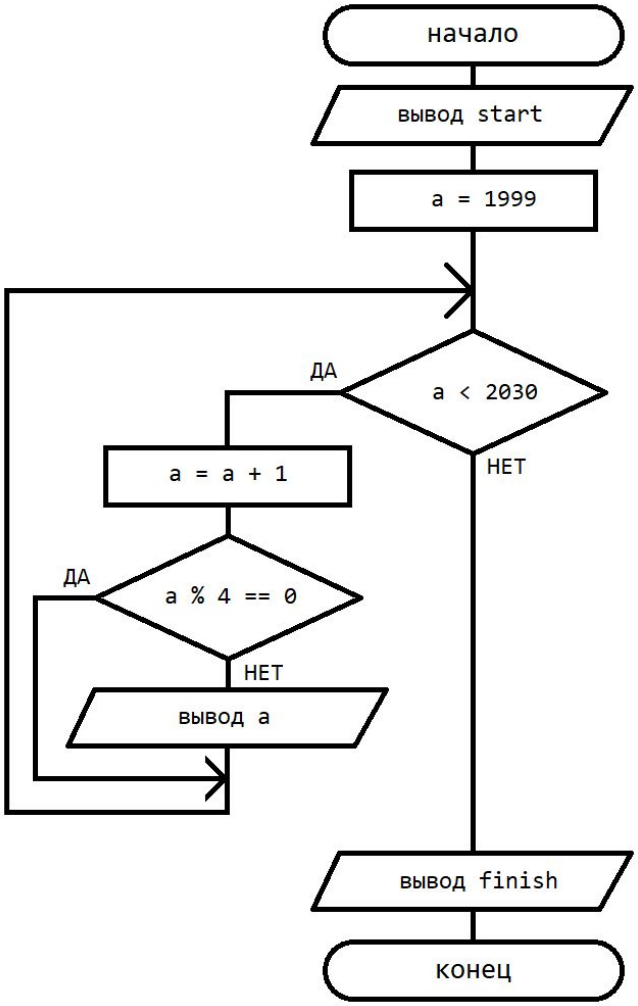
Вывод 2000 2001 2002 2003 … 2030 - только НЕВИСОКОСНЫЕ

```c
#include <stdio.h>

void main() {
    printf("main() start\n");

    int a = 1999;

    while (a < 2030) {
        a = a + 1;
        if (a % 4 == 0)
            goto label_body_end;
        printf("%d ", a);
    label_body_end:;
    }


    printf("\nmain() finish\n");
}
```

```c
#include <stdio.h>

void main() {
    printf("main() start\n");

    int a = 1999;
label_while:
    if (a < 2030) {
        a = a + 1;
        if (a % 4 == 0)
            goto label_body_end;
        printf("%d ", a);
    label_body_end:
        goto label_while;
    }

    printf("\nmain() finish\n");
}
```

WHILE & CONTINUE vs GOTO (3)

Вывод 2000 2001 2002 2003 ... 2030 - только НЕВИСОКОСНЫЕ

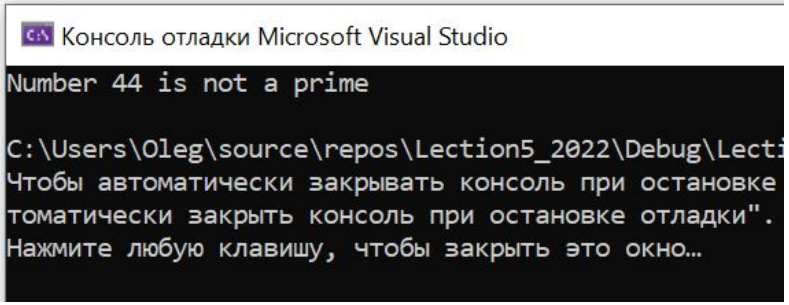



```
#include <stdio.h>

void main() {
    printf("main() start\n");

    int a = 1999;
label_while:
    if (a < 2030) {
        a = a + 1;
        if (a % 4 == 0)
            goto label_body_end;
        printf("%d ", a);
    label_body_end:
        goto label_while;
    }

    printf("\nmain() finish\n");
}
```

RETURN

Выясняем является ли число num простым (=prime) или является составным (=composite)

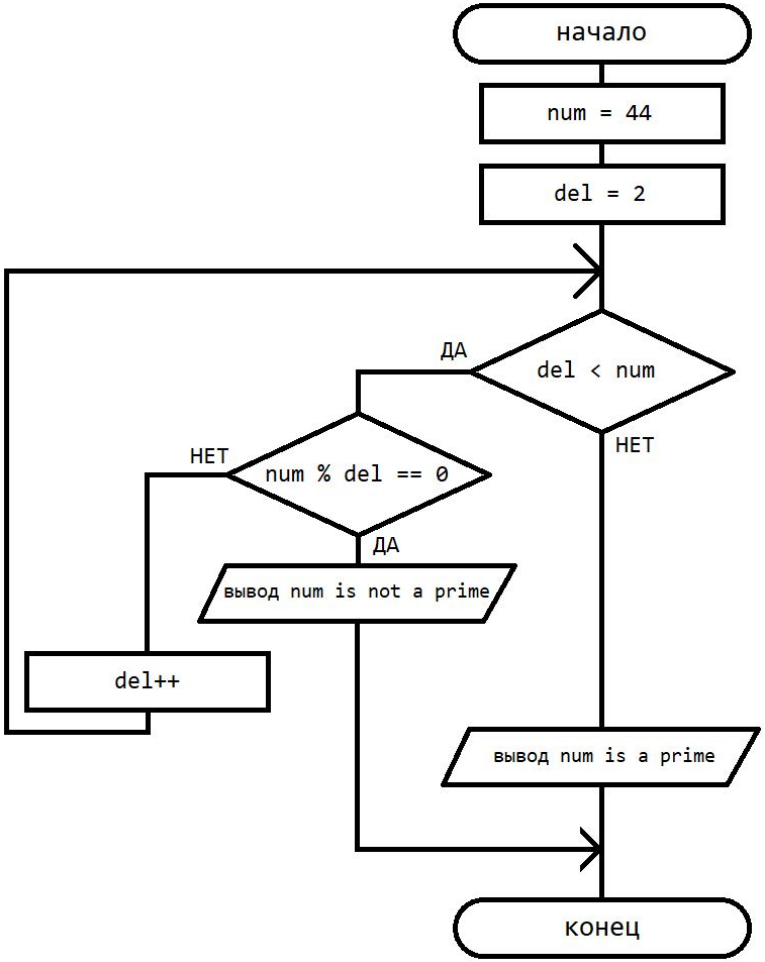
```
#include <stdio.h>

void main() {
    int num = 44;

    int del = 2;
    while (del < num) {
        // если num нацело делится на del
        if (num % del == 0) {
            // если есть иные делители, кроме 1 и num -
            // то число num не простое!
            printf("Number %d is not a prime\n", num);
            return;
        }
        del++;
    }

    printf("Number %d is a prime\n", num);
    return;
}
```



RETURN (2)

Выясняем является ли число num простым (=prime) или является составным (=composite)

```
#include <stdio.h>

void main() {
      int num = 44;

      int del = 2;
      while (del < num) {
            // если num нацело делится на del
            if (num % del == 0) {
                  // если есть иные делители, кроме 1 и num -
                  // то число num не простое!
                  printf("Number %d is not a prime\n", num);
                  return;
            }
            del++;
      }

      printf("Number %d is a prime\n", num);
      return;
}
```



RETURN (3)

Выясняем является ли число num простым (=prime) или является составным (=composite)

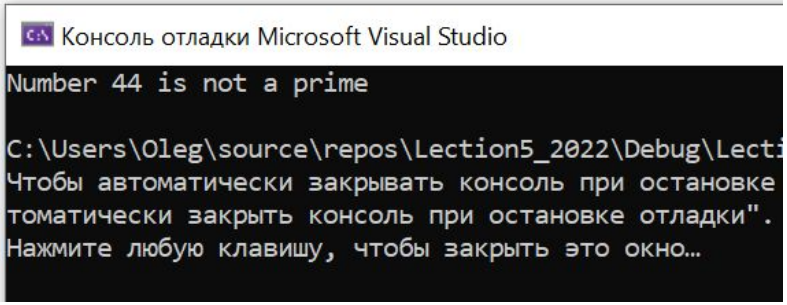
```
#include <stdio.h>

void main() {
    int num = 44;

    int del = 2;
    while (del < num) {
        // если num нацело делится на del
        if (num % del == 0) {
            // если есть иные делители, кроме 1 и num -
            // то число num не простое!
            printf("Number %d is not a prime\n", num);
            return;
        }
        del++;
    }

    printf("Number %d is a prime\n", num);
    // return;
}
```

Консоль отладки Microsoft Visual Studio

```
Number 44 is not a prime

C:\Users\Oleg\source\repos\Lection5_2022\Debug\Lecti
Чтобы автоматически закрывать консоль при остановке
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно…
```

RETURN vs GOTO (1)

Выясняем является ли число num простым (=prime) или является составным (=composite)

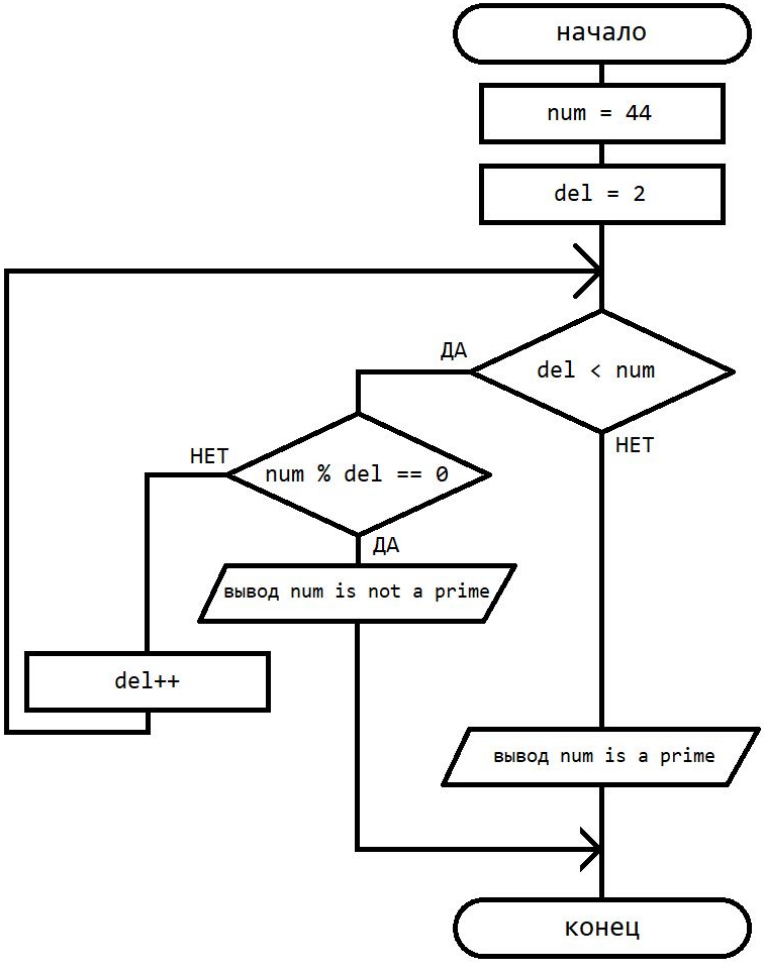



```
#include <stdio.h>

void main() {
    int num = 44;

    int del = 2;
    while (del < num) {
        // если num нацело делится на del
        if (num % del == 0) {
            // если есть иные делители, кроме 1 и num -
            // то число num не простое!
            printf("Number %d is not a prime\n", num);
            goto label_end;
        }
        del++;
    }

    printf("Number %d is a prime\n", num);
label_end:;
}
```

RETURN vs GOTO (2)

Выясняем является ли число num простым (=prime) или является составным (=composite)

```
#include <stdio.h>

void main() {
      int num = 44;

      int del = 2;
      while (del < num) {
            // если num нацело делится на del
            if (num % del == 0) {
                  // если есть иные делители, кроме 1 и num -
                  // то число num не простое!
                  printf("Number %d is not a prime\n", num);
                  goto label_end;
            }
            del++;
      }

      printf("Number %d is a prime\n", num);
label_end:;
}
```



RETURN со значением (1)

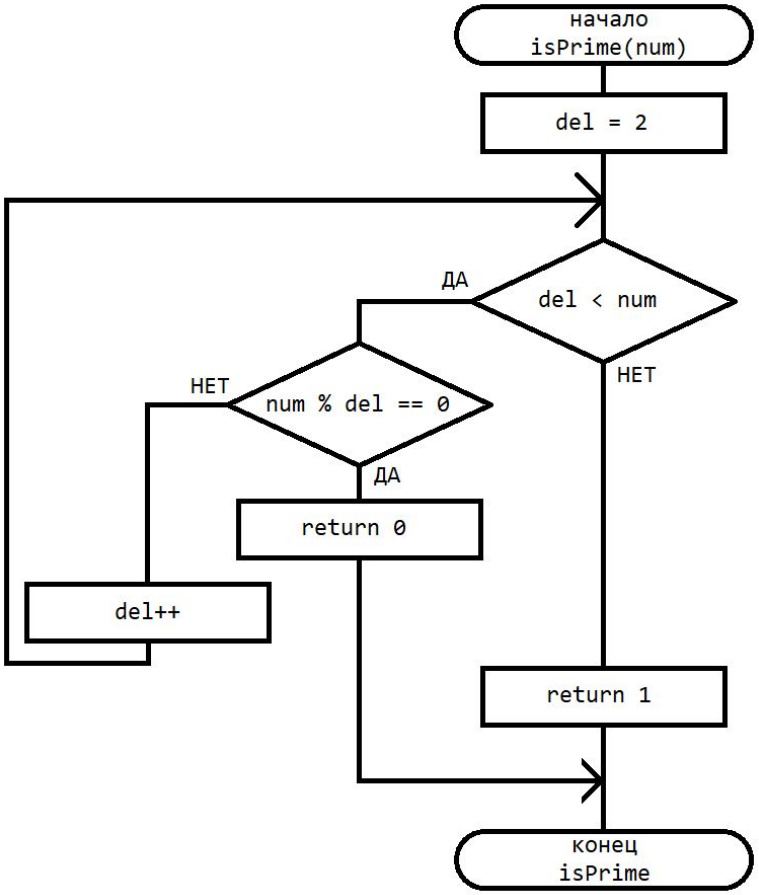
Выясняем является ли число num простым (=prime) или является составным (=composite). Реализация через отдельную функцию.

```
#include <stdio.h>

// Возвращает 1 - если число num "ПРОСТОЕ"
// (делится только на 1 и на само себя)
// Возвращает 0 в противном случае.
int isPrime(int num) {
    int del = 2;
    while (del < num) {
        // если num нацело делиться на del
        if (num % del == 0) {
            // возвращаем 0, т.к. число не простое
            return 0;
        }
        del++;
    }
    // возвращаем 1, т.к. ни одно число от 2 до num-1
    // не является делителем
    return 1;
}
```

RETURN со значением (1.1)

Выясняем является ли число num простым (=prime) или является составным (=composite). Реализация через отдельную функцию.

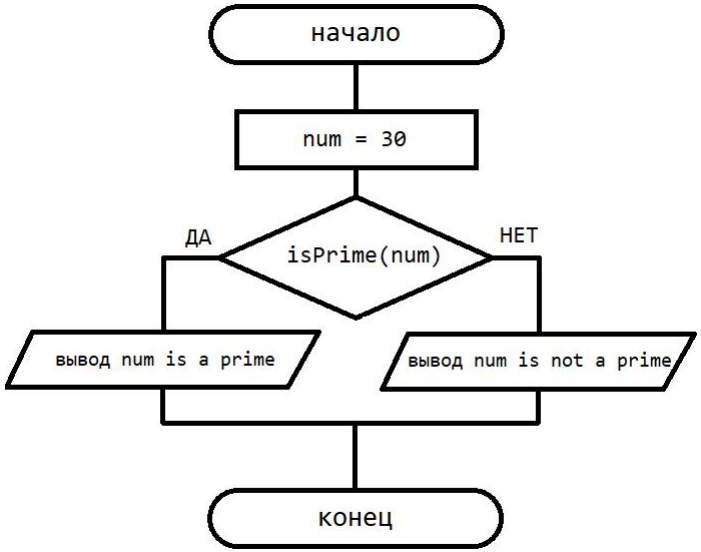
```
#include <stdio.h>

// Возвращает 1 - если число num "ПРОСТОЕ"
// (делится только на 1 и на само себя)
// Возвращает 0 в противном случае.
int isPrime(int num) {
      int del = 2;
      while (del < num) {
            // если num нацело делиться на del
            if (num % del == 0) {
                  // возвращаем 0, т.к. число не простое
                  return 0;
            }
            del++;
      }
      // возвращаем 1, т.к. ни одно число от 2 до num-1
      // не является делителем
      return 1;
}
```



RETURN со значением (2)

Выясняем является ли число num простым (=prime) или является составным (=composite). Реализация через отдельную функцию.

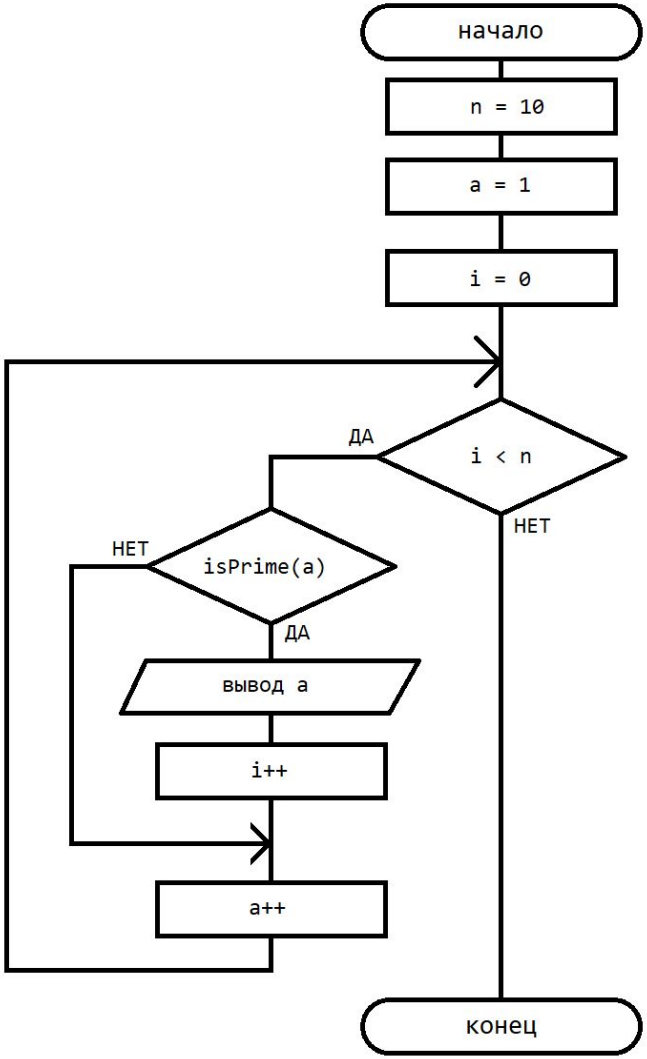
```
void main() {
    int num = 30;

    if (isPrime(num))
        printf("Number %d is a prime\n", num);
    else
        printf("Number %d is not a prime\n", num);
}
```



Вызов функции в WHILE

вывести N первых простых чисел

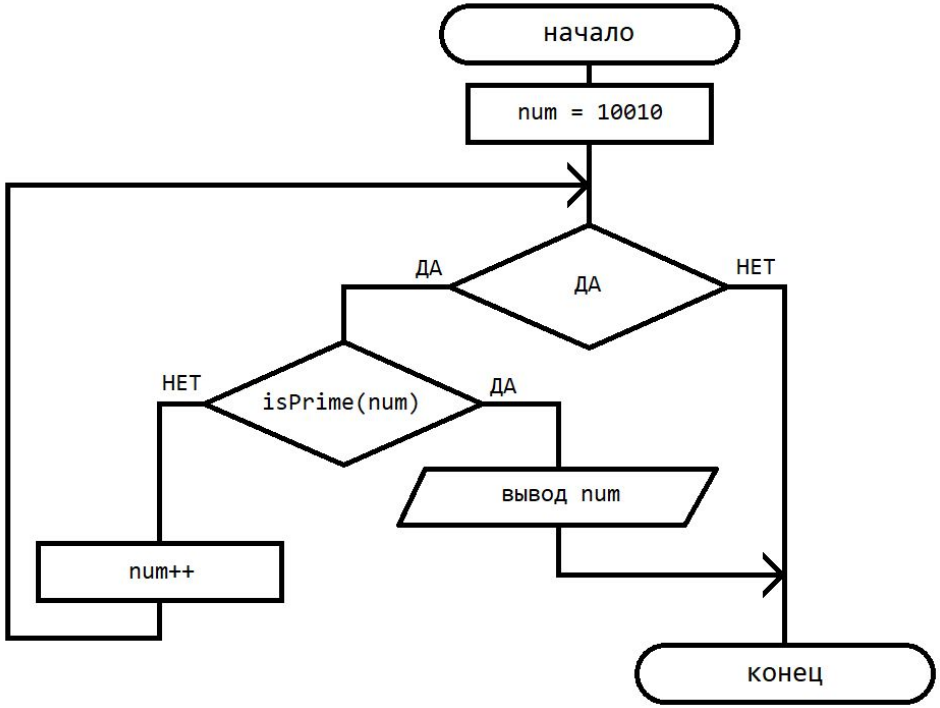
```
void main() {
    int n = 10;

    int a = 1;
    int i = 0;
    while (i < n) {
        if (isPrime(a)) {
            printf("%d ", a);
            i++;
        }
        a++;
    }
}
```



Бесконечный WHILE и выход из него

вывести ровно одно простое число не меньше заданного num

```
void main() {
    int num = 10010;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            break;
        }
        num++;
    }
}
```

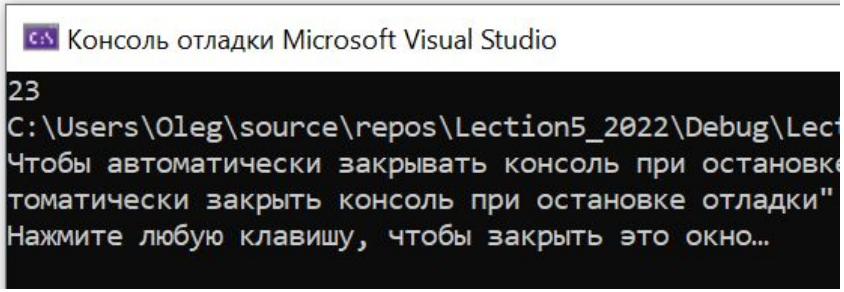


BREAK vs GOTO (1)

вывести ровно одно простое число не меньше заданного num

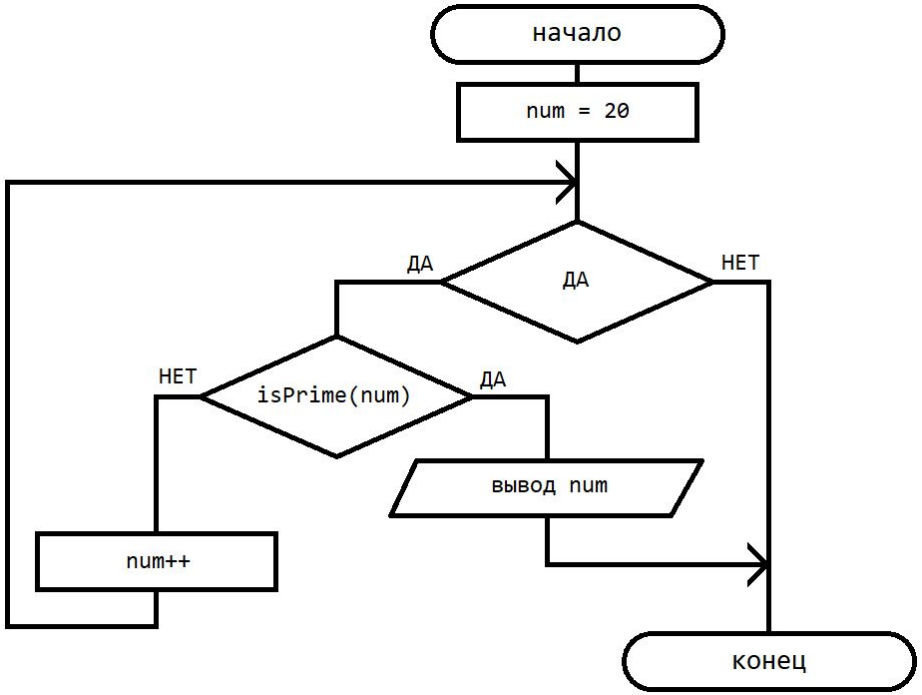
```
void main() {
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            break;
        }
        num++;
    }

}
```

```
void main() {
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            goto while_after;
        }
        num++;
    }
while_after:;
}
```

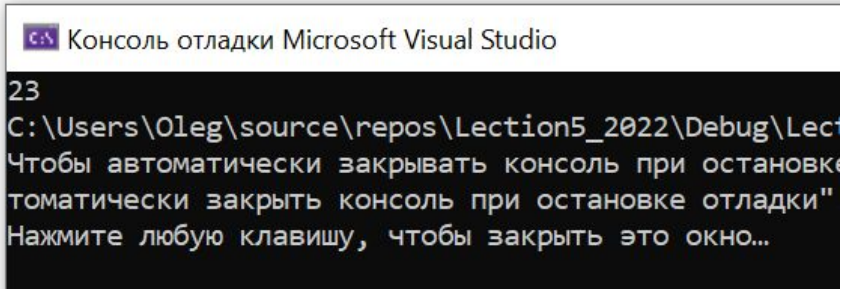


BREAK vs GOTO (2)

вывести ровно одно простое число не меньше заданного num




```
void main() {
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            goto while_after;
        }
        num++;
    }
while_after:;
}
```



BREAK vs RETURN (1)

вывести ровно одно простое число не меньше заданного num

```
void main() {
    printf("main() start\n");
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            break;
        }
        num++;
    }

    printf("\nmain() finish\n");
}
```

```
void main() {
    printf("main() start\n");
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            goto while_after;
        }
        num++;
    }
while_after:
    printf("main() finish\n");
}
```

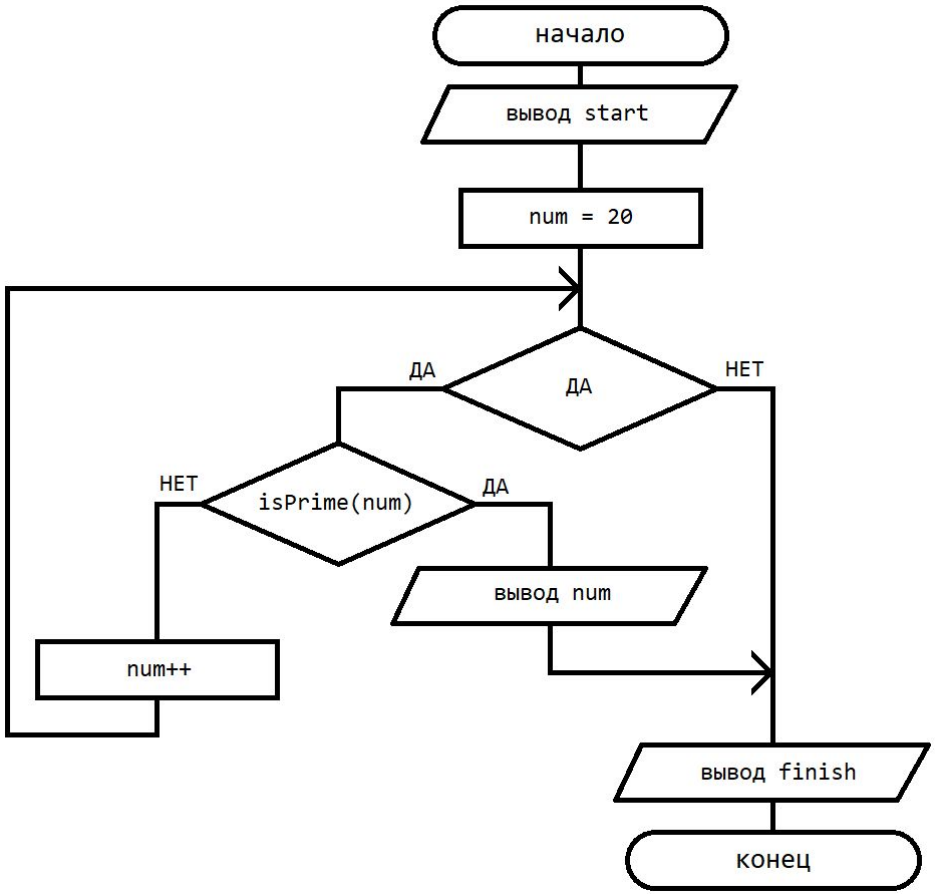
Консоль отладки Microsoft Visual Studio

```
main() start
23
main() finish

C:\Users\Oleg\source\repos\Lection5_202
```

BREAK vs RETURN (2)

вывести ровно одно простое число не меньше заданного num

```
void main() {
    printf("main() start\n");
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            break;
        }
        num++;
    }

    printf("\nmain() finish\n");
}
```



Консоль отладки Microsoft Visual Studio

```
main() start
23
main() finish

C:\Users\Oleg\source\repos\Lection5_202
```

BREAK vs RETURN (3)

вывести ровно одно простое число не меньше заданного num

```
void main() {
    printf("main() start\n");
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            break;
        }
        num++;
    }

    printf("\nmain() finish\n");
}
```

Консоль отладки Microsoft Visual Studio

```
main() start
23
main() finish

C:\Users\Oleg\source\repos\Lection5_202
```

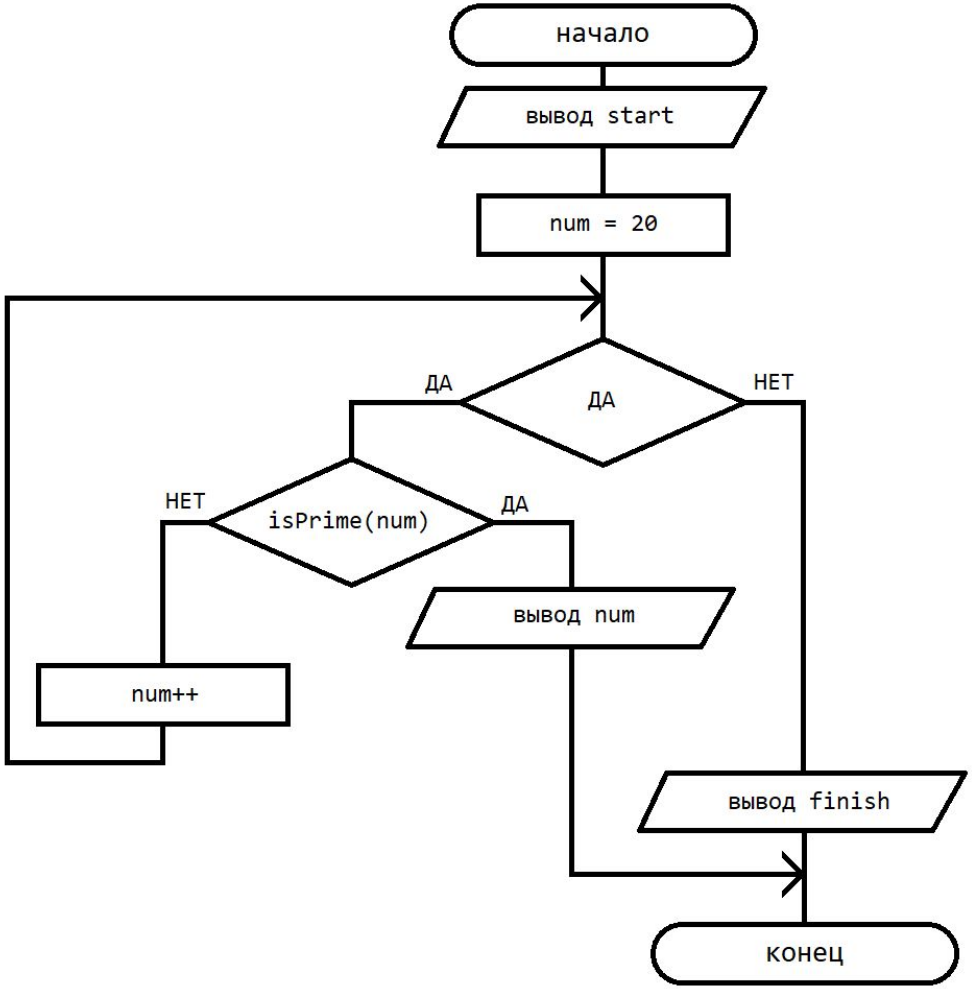
```
void main() {
    printf("main() start\n");
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            return;
        }
        num++;
    }

    printf("\nmain() finish\n");
}
```

Консоль отладки Microsoft Visual Studio

```
main() start
23
C:\Users\Oleg\source\repos\Lection
Чтобы автоматически закрывать конс
```

BREAK vs RETURN (4)

вывести ровно одно простое число не меньше заданного num




```
void main() {
    printf("main() start\n");
    int num = 20;

    while (1) {
        if (isPrime(num)) {
            printf("%d ", num);
            return;
        }
        num++;
    }

    printf("\nmain() finish\n");
}
```

Консоль отладки Microsoft Visual Studio

```
main() start
23
C:\Users\Oleg\source\repos\Lection
Чтобы автоматически закрывать конс
```

SWITCH (1)

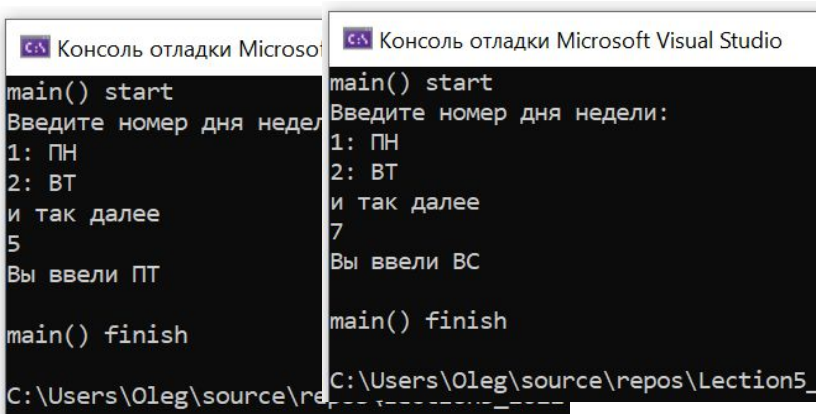
Ввести номер дня недели - вывести его название

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

void main() {
    printf("main() start\n");

    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int n;

    printf("Введите номер дня недели:\n");
    printf("1: ПН\n");
    printf("2: ВТ\n");
    printf("и так далее\n");

    scanf_s("%d", &n);
```

SWITCH (2)

Ввести номер дня недели - вывести его название

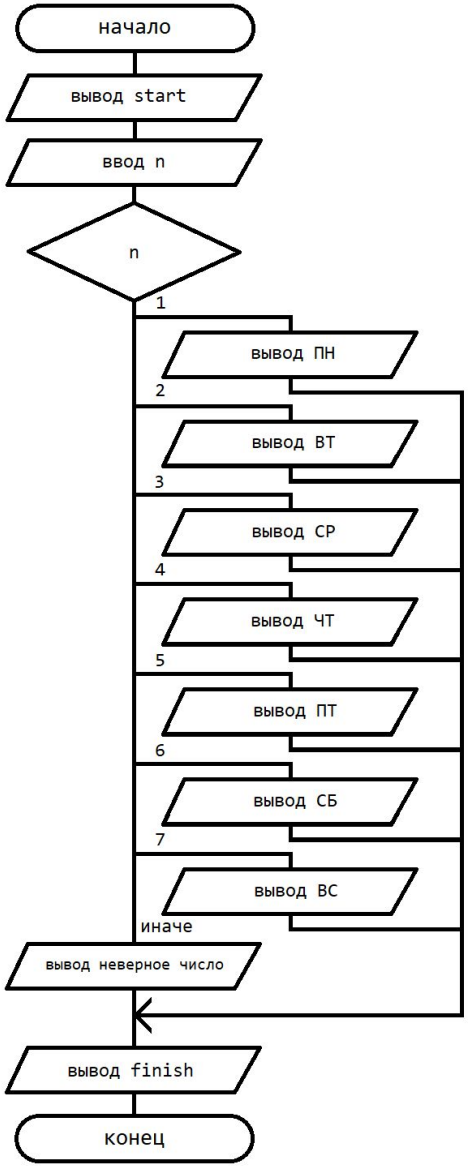
```
switch (n) {
case 1:
    printf("Вы ввели ПН\n");
    break;
case 2:
    printf("Вы ввели ВТ\n");
    break;
case 3:
    printf("Вы ввели СР\n");
    break;
case 4:
    printf("Вы ввели ЧТ\n");
    break;
case 5:
    printf("Вы ввели ПТ\n");
    break;
case 6:
    printf("Вы ввели СБ\n");
    break;
case 7:
    printf("Вы ввели ВС\n");
    break;
default:
    printf("Вы ввели неверное число\n");
    break;
}

printf("\nmain() finish\n");
}
```



SWITCH (3)

Ввести номер дня недели - вывести его название

```
        switch (n) {
        case 1:
                printf("Вы ввели ПН\n");
                break;
        case 2:
                printf("Вы ввели ВТ\n");
                break;
        case 3:
                printf("Вы ввели СР\n");
                break;
        case 4:
                printf("Вы ввели ЧТ\n");
                break;
        case 5:
                printf("Вы ввели ПТ\n");
                break;
        case 6:
                printf("Вы ввели СБ\n");
                break;
        case 7:
                printf("Вы ввели ВС\n");
                break;
        default:
                printf("Вы ввели неверное число\n");
                break;
        }

        printf("\nmain() finish\n");
}
```



SWITCH в Win приложении

```
124  LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)
125  {
126      switch (message)
127      {
128      case WM_COMMAND:
129          {
130              int wmId = LOWORD(wParam);
131              // Разобрать выбор в меню:
132              switch (wmId)
133              {
134              case IDM_ABOUT:
135                  DialogBox(hInst, MAKEINTRESOURCE(IDD_ABOUTBOX), hWnd, About);
136                  break;
137              case IDM_EXIT:
138                  DestroyWindow(hWnd);
139                  break;
140              default:
141                  return DefWindowProc(hWnd, message, wParam, lParam);
142              }
143          }
144          break;
145      case WM_PAINT:
146          {
147              PAINTSTRUCT ps;
148              HDC hdc = BeginPaint(hWnd, &ps);
149              // TODO: Добавьте сюда любой код прорисовки, использующий HDC...
150              EndPaint(hWnd, &ps);
151          }
152          break;
153      case WM_DESTROY:
154          PostQuitMessage(0);
155          break;
156      default:
157          return DefWindowProc(hWnd, message, wParam, lParam);
158      }
159      return 0;
160  }
```

SWITCH в Win приложении (2)

```
128     case WM_COMMAND:
129         {
130             int wmId = LOWORD(wParam);
131             // Разобрать выбор в меню:
132             switch (wmId)
133             {
134             case IDM_ABOUT:
135                 DialogBox(hInst, MAKEINTRESOURCE(IDD_ABOUTBOX), hWnd, About);
136                 break;
137             case IDM_EXIT:
138                 DestroyWindow(hWnd);
139                 break;
140             default:
141                 return DefWindowProc(hWnd, message, wParam, lParam);
142             }
143         }
144         break;
```

Реализация консольного меню при помощи SWITCH (1)

Заготовка для ЛР10 и ЛР11 – меню выбора действия пользователя (консоль)

```
1   #include <stdio.h>
2   #include <Windows.h>
3
4   void print1_10() {
5       int a = 1;
6       do {
7           printf("%d ", a);
8           a += 1;
9       } while (a <= 10);
10  }
11
12  void print10_1() {
13      int a = 10;
14      do {
15          printf("%d ", a);
16          a -= 1;
17      } while (a >= 1);
18  }
19
20  void print5Odds() {
21      int a = 1;
22      int i = 1;
23      do {
24          printf("%d ", a);
25          a += 2;
26          i += 1;
27      } while (i <= 5);
28  }
29
```

Реализация консольного меню при помощи SWITCH (2)

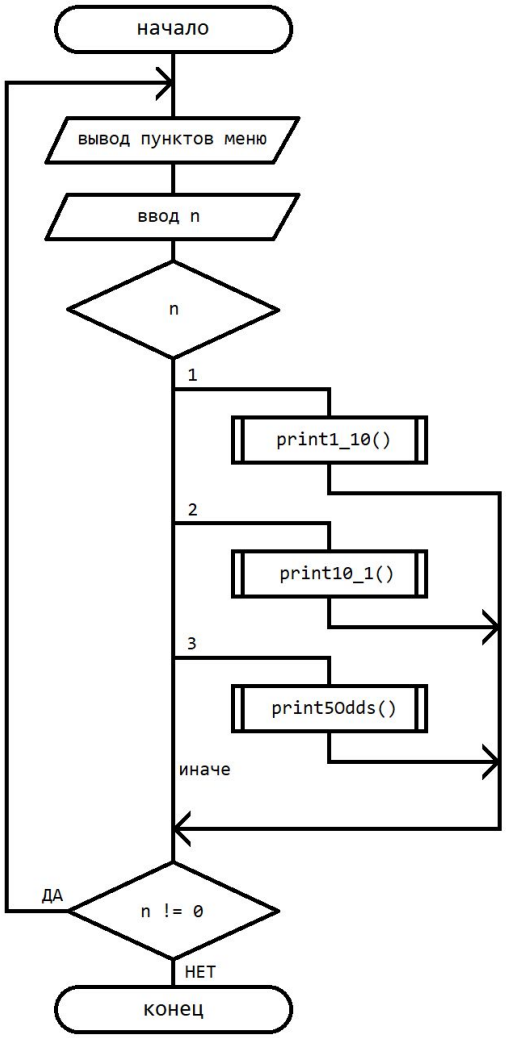
Заготовка для ЛР10 и ЛР11 – меню выбора действия пользователя (консоль)

```
void main() {
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int n;
    do {
        printf("\n");
        printf("\n");
        printf("Выберите нужную вам операцию:\n");
        printf("1: Вывести числа от 1 до 10\n");
        printf("2: Вывести числа от 10 до 1\n");
        printf("3: Вывести 5 первых нечетных чисел начиная с 1\n");
        printf("\n");
        printf("0: Выйти из программы\n");

        scanf_s("%d", &n);
```

Реализация консольного меню при помощи SWITCH (3)

Заготовка для ЛР10 и ЛР11 – меню выбора действия пользователя (консоль)

```
        switch (n) {
        case 1:
            print1_10();
            break;

        case 2:
            print10_1();
            break;

        case 3:
            print5Odds();
            break;

        }

    } while (n != 0);
}
```



Лабораторная работа №10

Консольное меню и циклические задачи (DO WHILE)

Задача 1. Сделать меню для управления

Программа должна выводить числа от 1 до 10 - если пользователь выбрал '1'
Программа должна выводить числа от 10 до 1 - если пользователь выбрал '2'
Программа должна выводить 5 первых нечетных чисел - если пользователь выбрал '3'
Программа должна завершаться - если пользователь выбрал '0'

```c
#include <stdio.h>
#include <Windows.h>

void print1_10() {
    int a = 1;
    do {
        printf("%d ", a);
        a += 1;
    } while (a <= 10);
}

void print10_1() {
    int a = 10;
    do {
        printf("%d ", a);
        a -= 1;
    } while (a >= 1);
}

void print5Odds() {
    int a = 1;
    int i = 1;
    do {
        printf("%d ", a);
        a += 2;
        i += 1;
    } while (i <= 5);
}
```

```c
void main() {
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int n;
    do {
        printf("\n");
        printf("\n");
        printf("Выберите нужную вам операцию:\n");
        printf("1: Вывести числа от 1 до 10\n");
        printf("2: Вывести числа от 10 до 1\n");
        printf("3: Вывести 5 первых нечетных чисел начиная с 1\n");
        printf("\n");
        printf("0: Выйти из программы\n");

        scanf_s("%d", &n);
        switch (n) {
        case 1:
            print1_10();
            break;
        case 2:
            print10_1();
            break;
        case 3:
            print5Odds();
            break;
        }

    } while (n != 0);
}
```

Задача 2. Добавить пункт меню ‘4’

Если пользователь выбрал ‘4’, то нужно вывести 10 первых четных чисел.

```
C:\Users\Oleg\source\repos\Labs\Debug\Lab10.exe

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1

0: Выйти из программы
3
1 3 5 7 9

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1

0: Выйти из программы
4
2 4 6 8 10 12 14 16 18 20

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1

0: Выйти из программы
```

```c
void print10Evens() {
    int a = 2;
    int i = 1;
    do {
        printf("%d ", a);
        a += 2;
        i += 1;
    } while (i <= 10);
}
```

```c
        printf("\n");
        printf("\n");
        printf("Выберите нужную вам операцию:\n");
        printf("1: Вывести числа от 1 до 10\n");
        printf("2: Вывести числа от 10 до 1\n");
        printf("3: Вывести 5 первых нечетных чисел начиная с 1\n");
        printf("4: Вывести 10 первых четных чисел начиная с 1\n");
        printf("\n");
        printf("0: Выйти из программы\n");

        scanf_s("%d", &n);

        switch (n) {
        case 1:
            print1_10();
            break;

        case 2:
            print10_1();
            break;

        case 3:
            print5Odds();
            break;

        case 4:
            print10Evens();
            break;
        }
```

Задача 3. Добавить пункт меню ‘5’

Если пользователь выбрал ‘5’, то нужно вывести 10 первых чисел ряда 10 20 30 … .

C:\Users\Oleg\source\repos\Labs\Debug\Lab10.exe

```
Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...

0: Выйти из программы
5
10 20 30 40 50 60 70 80 90 100

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...

0: Выйти из программы
```

```
40  void print10_100() {
41      int a = 10;
42      int i = 1;
43      do {
44          printf("%d ", a);
45          a += 10;
46          i += 1;
47      } while (i <= 10);
48  }
```

Задача 4. Добавить пункт меню ‘6’

Если пользователь выбрал ‘6’, то нужно вывести 10 первых чисел ряда 100 200 300 … .

```
C:\Users\Oleg\source\repos\Labs\Debug\Lab10.exe

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...

0: Выйти из программы
6
100 200 300 400 500 600 700 800 900 1000

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...

0: Выйти из программы
```

Задача 5*. Добавить пункт меню ‘7’

Если пользователь выбрал ‘7’, то нужно вывести 11 первых чисел ряда 1 2 4 8 16 32 … .

C:\Users\Oleg\source\repos\Labs\Debug\Lab10.exe

```
Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...
7: Вывести 11 первых чисел ряда 1 2 4 8 16 32  ...

0: Выйти из программы
7
1 2 4 8 16 32 64 128 256 512 1024

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...
7: Вывести 11 первых чисел ряда 1 2 4 8 16 32  ...

0: Выйти из программы
```

Домашнее задание по ЛР10

1) Доделать задачи 1-4.
2) Добавить в меню пункт «Вывести 11 первых чисел ряда 1 3 9 27 81 … » (каждый следующий больше предыдущего в 3 раза
3) * Каждую задачу из ЛР6 (цикл DO WHILE) добавить как отдельный пункт меню.
4) ** Реализовать две-три задачи вместо DO WHILE через IF GOTO
5) Обязательно! Принести получившийся код на занятие. Его будем использовать и переделывать на следующих лабораторных работах.

ИТОГО по ЛР10

1. Научились создавать меню.

Лабораторная работа №11

Цикл WHILE

Задача 1. Добавить вычисления через WHILE

Продолжаем работать с кодом, созданным в ЛР10.
Если пользователь выбрал ‘11’, то вывести числа от 100 до 10 с шагом 10.
Реализовать это вычисление через WHILE.

C:\Users\Oleg\source\repos\Labs\Debug\Lab10.exe

```
Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...
7: Вывести 11 первых чисел ряда 1 2 4 8 16 32  ...
11: Вывести числа 100 90 80 ... 10 (через WHILE)

0: Выйти из программы
11
100 90 80 70 60 50 40 30 20 10

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...
7: Вывести 11 первых чисел ряда 1 2 4 8 16 32  ...
11: Вывести числа 100 90 80 ... 10 (через WHILE)

0: Выйти из программы
```

```c
void print100_10_while() {
    int a = 100;
    while (a >= 10) {
        printf("%d ", a);
        a -= 10;
    }
}
```

Задача 2. 1000 900 800 … 100

Добавить пункт меню ‘12’. Если пользователь выбрал ‘12’, то вывести числа от 1000 до 100 с шагом 100. Реализовать это вычисление через WHILE.

```
C:\Users\Oleg\source\repos\Labs\Debug\Lab10.exe

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...
7: Вывести 11 первых чисел ряда 1 2 4 8 16 32  ...
11: Вывести числа 100 90 80 ... 10 (через WHILE)
12: Вывести числа 1000 900 800 ... 100 (через WHILE)

0: Выйти из программы
12
1000 900 800 700 600 500 400 300 200 100

Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...
7: Вывести 11 первых чисел ряда 1 2 4 8 16 32  ...
11: Вывести числа 100 90 80 ... 10 (через WHILE)
12: Вывести числа 1000 900 800 ... 100 (через WHILE)

0: Выйти из программы
```

Задача 3. Все пункты с 1 до 10 переделать в WHILE

Переделать все функции, реализованные через DO WHILE в WHILE.
Проверить, что все пункты меню корректно работают.

Задача 4*. Рост суммы на счете

Добавить пункт меню ‘13’. Если пользователь выбрал ‘13’, то вывести состояние счета, на котором изначально было S рублей, при ставке N% годовых. Нужно вывести изменение счета за 10 лет.

```
C:\Users\Oleg\source\repos\Labs\Debug\Lab10.exe
0: Выйти из программы
13
s = 10000, n = 10
через 0 лет: 10000
через 1 лет: 11000
через 2 лет: 12100
через 3 лет: 13310
через 4 лет: 14641
через 5 лет: 16105
через 6 лет: 17715
через 7 лет: 19486
через 8 лет: 21434
через 9 лет: 23577
через 10 лет: 25934


Выберите нужную вам операцию:
1: Вывести числа от 1 до 10
2: Вывести числа от 10 до 1
3: Вывести 5 первых нечетных чисел начиная с 1
4: Вывести 10 первых четных чисел начиная с 1
5: Вывести 10 первых чисел ряда 10 20 30 ...
6: Вывести 10 первых чисел ряда 100 200 300 ...
7: Вывести 11 первых чисел ряда 1 2 4 8 16 32  ...
11: Вывести числа 100 90 80 ... 10 (через WHILE)
12: Вывести числа 1000 900 800 ... 100 (через WHILE)
13: Вывести состояние счета

0: Выйти из программы
```

Домашнее задание ЛР11

1) Доделать задачи 1-3, которые не успели сделать на занятии в классе.
2) Добавьте пункт меню ‘20’ выводящий числа от 1000 до 0 с шагом N. Реализовать вычисление через WHILE. N = номеру вашего варианта/порядкового номера в журнале.
3) *Добавьте пункт меню ‘21’, выводящий N первых факториалов (1 2 6 24 120 720 …). N вводится с клавиатуры.
4) ** Добавить пункт при помощи которого вывести первые N чисел Фибоначчи. N вводится с клавиатуры.
5) ** Добавить пункт при помощи которого вывести разложение числа N на простые множители. N вводится с клавиатуры.
6) ** Несколько задач вместо WHILE реализовать через IF GOTO

ИТОГО по ЛР11

1. Познакомились с WHILE

ИТОГО по лекции 6

1. Узнали почти всё про управление в Си – повторили про IF, DO WHILE, узнали про WHILE, CONTINUE, BREAK, RETURN, GOTO, SWITCH
2. Узнали чем плох GOTO, и почему его не стоит использовать в ваших программах
3. Узнали как создать меню в консоли
4. Узнали что нужно сделать в ЛР10 и ЛР11