



- 6 Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальное описание с пред- и постусловиями)

**ЛР №10:**

List – вывод пакета программы

Break – задача точки остановки на указанной строке

Run – запуск программы на выполнение

Next – выполняется следующая строка

Print – печать значения выражения, включающая переменные и вызовы функций

Set var – присваивание значение переменной

Step – выполнение следующего шага

Bt – распечатка содержимого стека вызова

Continue – продолжение работы программы после остановки

Quit – выход из отладчика

**ЛР №12:**

Вводится натуральное число, образовать из него три новых числа: первое состоит из цифр исходного числа кратных трем, второе число из цифр, которые при делении на 3 дают остаток 1, а третье из цифр исходного числа, которые при делении на 3 дают остаток 2. (Если в исходном числе нет цифр любого из трех указанных видов, то вместо числа, соответствующего отсутствующим цифрам, выдать 0). Распечатать три полученных числа и их сумму.

- 7 Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты, либо соображения по тестированию].

**ЛР №10:**

```
#include <stdio.h>
int main(){
    int a,b,c;
    a = 5, b=12, c = 3;
    int x = c*a*a*b;
    printf("%u\n",b*a);
    printf("%u\n",b*a);
    printf("%u\n",x%7);
}
gdb -c a.out.core a.out
list
break 3
run
next
next
next
next
print b*a
set var a=17
next
ptype c
bt
step
continue
quit
```

**ЛР №12:**

```

#include <stdio.h>
int reverse(int a){
    int b = 0;
    while(a>0){
        int c = a%10;
        b = b*10 + c;
        a /= 10;
    }
    return b;
}
int main(){
    int n, s1 = 0, s2 = 0, s3 = 0;
    scanf("%i", &n);
    while (n>0){
        int a = (n%10);
        if (a%3==0){
            s1 = s1*10 + a;
        }
        if (a%3==1){
            s2 = s2*10 + a;
        }
        if (a%3==2){
            s3 = s3*10 + a;
        }
        n /= 10;
    }
    s1 = reverse(s1);
    s2 = reverse(s2);
    s3 = reverse(s3);
    printf("-----\nПервое число: %d;\nВторое число: %d;\nТретье число: %d;\nСумма чисел: %d;\n", s1, s2, s3,
s1+s2+s3);
    return 0;
}

```

Допущен к выполнению работы. Подпись преподавателя \_\_\_\_\_

## 8 Распечатка протокола (подклеить листинг окончательного варианта программы с текстовыми примерами, подписанный преподавателем)

```
(base) ramil@ramil:~/labs/labs 10,12$ cat header.txt
*****
*           Лабагорная работа №7           *
* Программирование в алгоритмической модели Маркова *
*      Выполнил студент гр. М8О-105-Б      *
*      Титесев Рамиль Маратович           *
*****
(base) ramil@ramil:~/labs/labs 10,12$ cat > test.c
#include <stdio.h>
int main(){
int a,b,c;
a = 5, b=12, c = 3;
int x = c*a*a*b;
printf("%u\n",b*a);
printf("%u\n",b*a);
printf("%u\n",x%7);
}
(base) ramil@ramil:~/labs/labs 10,12$ gcc -g -o a.out test.c
(base) ramil@ramil:~/labs/labs 10,12$ ./a.out
60
2
4
(base) ramil@ramil:~/labs/labs 10,12$ gdb -c a.out.core a.out
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from a.out...
/home/ramil/labs/labs 10,12/a.out.core: Нет такого файла или каталога.
(gdb) list
1 #include <stdio.h>
2
3 int main(){
4
5 int a,b,c;
6 a = 5, b=12, c = 3;
7 int x = c*a*a*b;
8 printf("%u\n",b*a);
9 printf("%u\n",b*a);
10 printf("%u\n",x%7);
(gdb) break 3
Breakpoint 1 at 0x1149: file test.c, line 3.
(gdb) run
Starting program: /home/ramil/labs/labs 10,12/a.out

Breakpoint 1, main () at test.c:3
3 int main(){
(gdb) next
6 a = 5, b=12, c = 3;
(gdb) next
7 int x = c*a*a*b;
(gdb) next
8 printf("%u\n",b*a);
(gdb) print b*a
$1 = 60
(gdb) set var a = 17
(gdb) next
```

```

204
9 printf("%u\n",b%a);
(gdb) ptype c
type = int
(gdb) bt
#0 main () at test.c:9
(gdb) step
__printf (format=0x555555556004 "%u\n") at printf.c:28
28 printf.c: Нет такого файла или каталога.
(gdb) continue
Continuing.
12
4
[Inferior 1 (process 6251) exited normally]
(gdb) quit

```

```

(base) ramil@ramil:~/labs/labs 10,12$ cat header.txt
*****
*           Лабараторная работа №10           *
* Техника работы с целыми числами. Система счисления *
*           Выполнил студент гр. М8О-105-Б           *
*           Титеев Рамиль Маратович                 *
*****

```

```

(base) ramil@ramil:~/labs/labs 10,12$ cat > lab12_v15.c
#include <stdio.h>
int reverse(int a){
    int b = 0;
    while(a>0){
        int c = a%10;
        b = b*10 + c;
        a /= 10;
    }
    return b;
}
int main(){
    int n, s1 = 0, s2 = 0, s3 = 0;
    scanf("%i", &n);
    while (n>0){
        int a = (n%10);
        if (a%3==0){
            s1 = s1*10 + a;
        }
        if (a%3==1){
            s2 = s2*10 + a;
        }
        if (a%3==2){
            s3 = s3*10 + a;
        }
        n /= 10;
    }
    s1 = reverse(s1);
    s2 = reverse(s2);
    s3 = reverse(s3);
    printf("-----\nПервое число: %d;\nВторое число: %d;\nТретье число: %d;\nСумма чисел: %d;\n", s1, s2, s3, s1+s2+s3);
    return 0;
}

```

```

(base) ramil@ramil:~/labs/labs 10,12$ g++ lab12_v15.c
(base) ramil@ramil:~/labs/labs 10,12$ ./a.out
11233455

```

```

-----
Первое число: 33;
Второе число: 114;
Третье число: 255;
Сумма чисел: 402;
(base) ramil@ramil:~/labs/labs 10,12$ ./a.out
223110231

```

```

-----
Первое число: 303;
Второе число: 111;
Третье число: 222;
Сумма чисел: 636;
(base) ramil@ramil:~/labs/labs 10,12$ ./a.out
72441501
-----

```

Первое число: 0;  
 Второе число: 74411;  
 Третье число: 25;  
 Сумма чисел: 74436;  
 (base) ramil@ramil:~/labs/labs 10,12\$ ./a.out  
 33693

-----  
 Первое число: 33693;  
 Второе число: 0;  
 Третье число: 0;  
 Сумма чисел: 33693;

- 9 **Дневник отладки** должен содержать дату и время сеансов отладки, и основные ошибки (ошибки в сценарии и программе, не стандартные операции) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание

- 10 Замечание автора по существу работы \_\_\_\_\_

- 11 Выводы \_\_\_\_\_ Я научился проверять пошагово программы, а также писать программы с целочисленным значением переменных.

Недочеты, допущенные при выполнении задания, могут быть устранены следующим образом Больше практики и чтение дополнительного материала

Подпись студента \_\_\_\_\_