

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Департамент программной инженерии
Дисциплина: «Архитектура вычислительных систем»

**ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ АССЕМБЛЕРА.
МИКРОПРОЕКТ.**

Пояснительная записка

Выполнил:
Мунтян Александр Александрович
студент гр. БПИ198.

Москва
2020

Содержание

1. Текст задания	2
2. Применяемые расчетные методы	3
2.1. Теория решения задания.....	3
2.2. Дополнительный функционал программы	Error! Bookmark not defined.
3. Тестирование программы.....	4
3.1. Корректные значения	4
3.2. Некорректные значения.....	6
ПРИЛОЖЕНИЕ 1. Список литературы	7
ПРИЛОЖЕНИЕ 2. Код программы.....	8

1. Текст задания

Разработать программу, определяющую число
непересекающихся повторов троек битов '011' в заданном
машинном слове

2. Применяемые расчетные методы

2.1. Теория решения задания

Считываем число и загружаем в `eax`. Циклом проходимся по машинному слову(в `edx` хранится индекс позиции (31 до 0).), в ходе каждой итерации текущий бит проверяется на равенство определенному биту из тройки(индекс бита из тройки хранится в `ebx`).

В случае равенства:

- 1) Если текущий бит проверялся на равенство последнему из тройки - увеличиваем счетчик количества найденных "011" и обнуляем `ebx`.
- 2) Иначе просто увеличиваем `ebx`.

После всех проверок уменьшаем `edx` на 1. Если он равен -1 -выводим результат.

3. Тестирование программы

3.1. Корректные значения

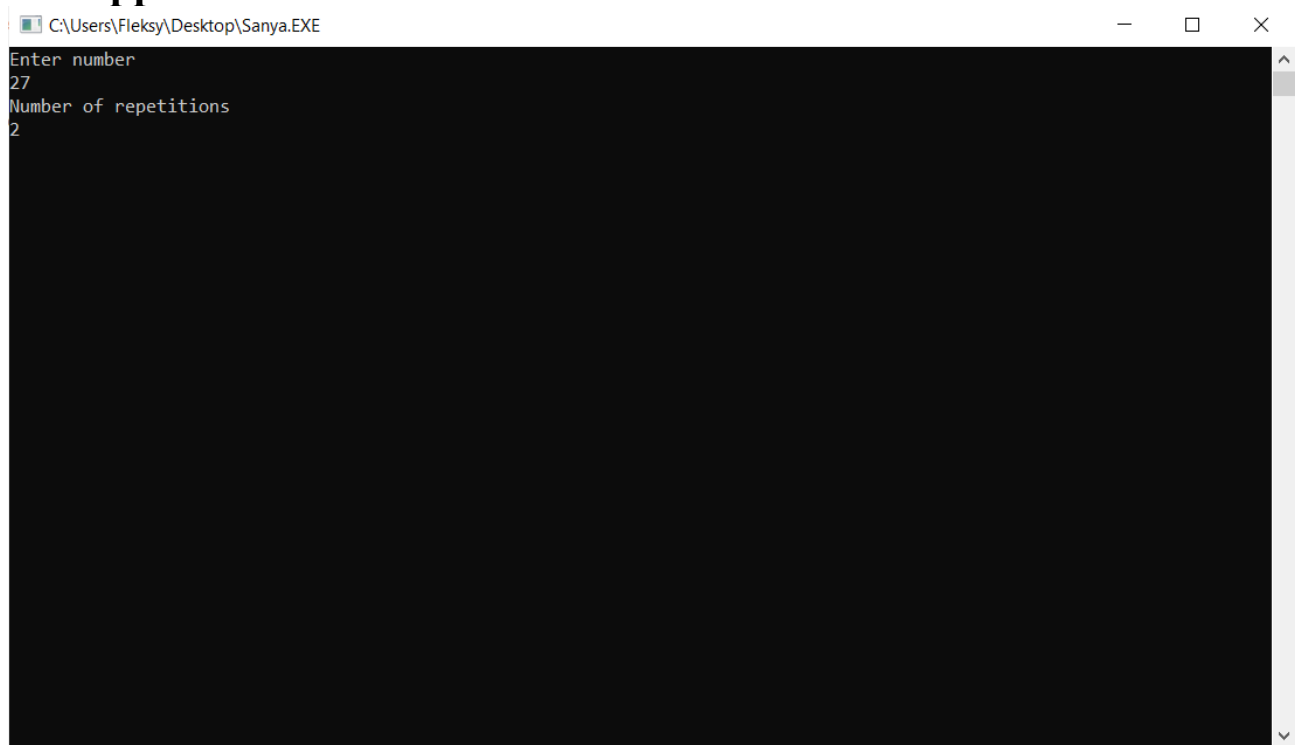


Рисунок 1. 27 представляется как 0..011011, т.е. в нем две таких тройки битов

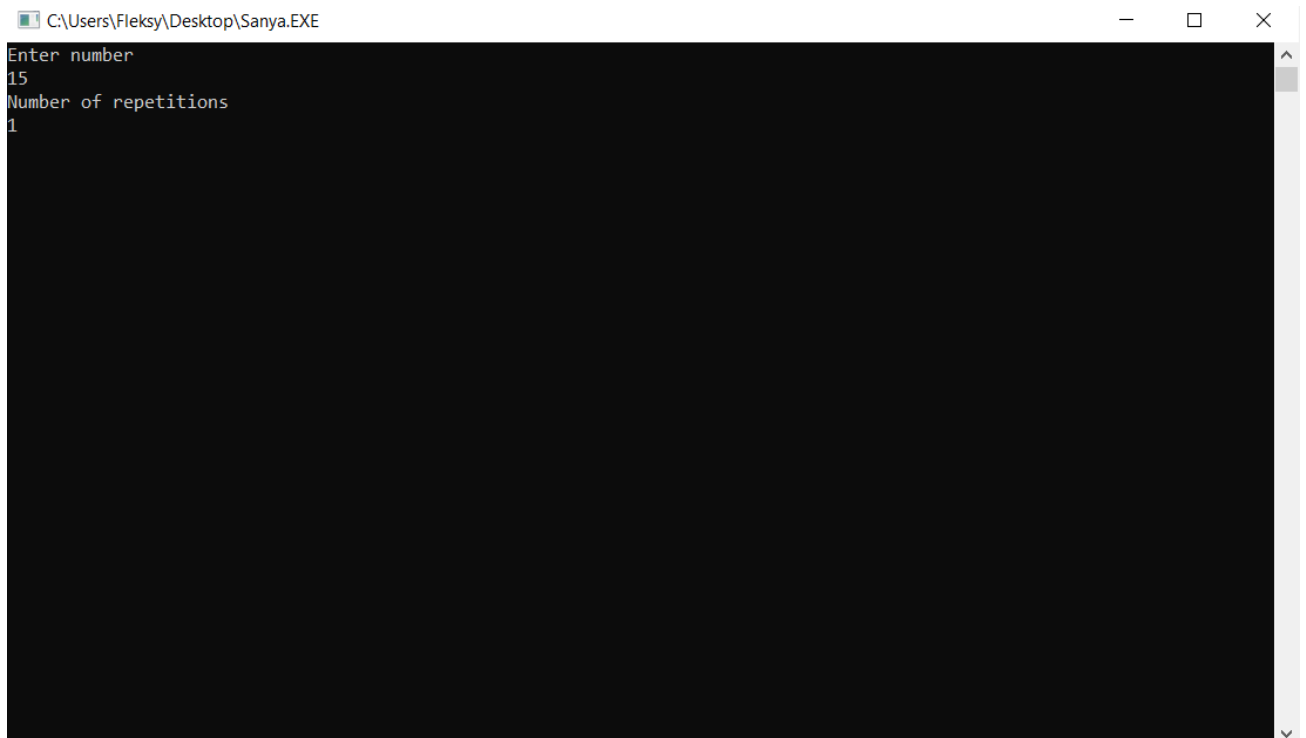


Рисунок 2. 15 представляется как 0..01111, т.е. в нем такая тройка битов одна

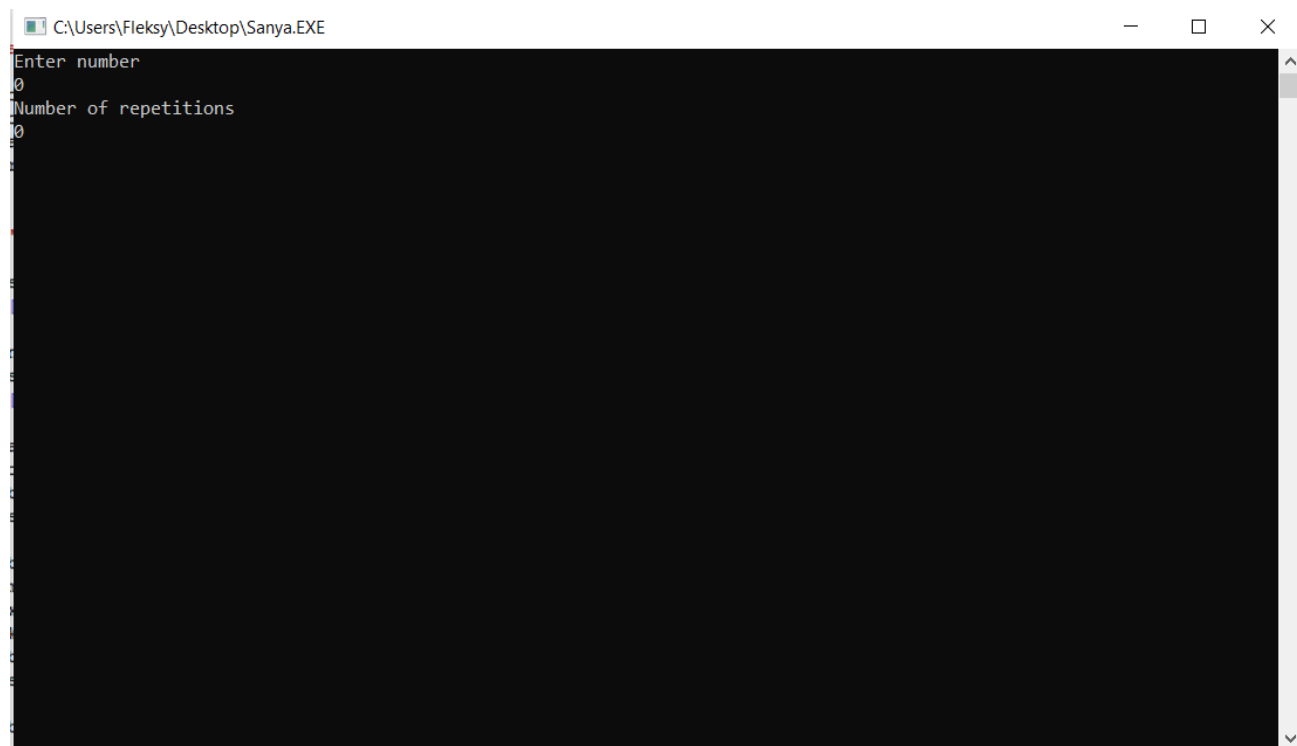


Рисунок 3. 0 представляется как 0..00, т.е. в нем нет таких тройки битов

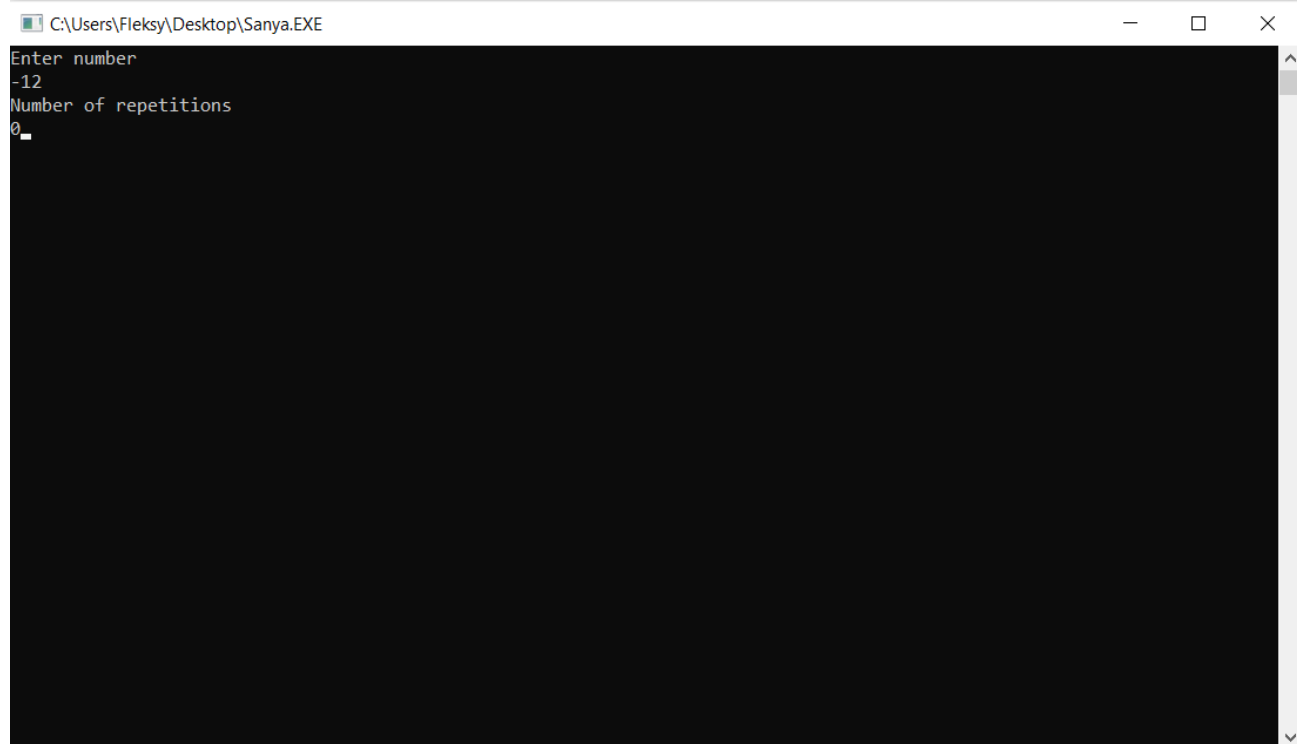


Рисунок 4. -12 представляется как 1..111110100, т.е. в нем таких троек битов ноль

3.2. Некорректные значения

Программа обрабатывает все числа. В случае если введено не число – возвращает 0.

Список литературы

1. SoftCraft разноликое программирование «Программирование на языке ассемблера. Микропроект. Требования к оформлению. 2020-2021 уч.г.» [Электронный ресурс] <http://softcraft.ru/edu/comparch/tasks/mp01/> (дата обращения: 01.11.2020)
2. «Программирование на языке ассемблера» [Электронный ресурс] <http://natalia.appmat.ru/c&c++/assembler.html> (дата обращения: 30.10.2020)

Код программы

1. format PE console
2. entry start
- 3.
4. include 'win32a.inc'
- 5.
6. section '.data' data readable writable
- 7.
8. str1 db 'Enter number ',10,0
9. str2 db 'Number of repetitions ',10,0
10. strScanInt db '%d',0 ;строка для ввода с помощью scanf
11. number dd ? ;переменная для введенного числа
12. count dd ? ;переменная для ответа
- 13.
- 14.section 'text' code executable readable
- 15.start:
16. push str1 ;Просим пользователя ввести число
17. call [printf]
- 18.
19. push number ;ввод
20. push strScanInt
21. call [scanf]
- 22.
23. mov eax,[number] ;загружаем в eax
24. mov edx,31 ;индекс проверяемого бита (с 31-го до 0-го)
25. xor ebx,ebx ;показывает, на совпадение с каким битом из тройки мы его проверяем

26.	xor esi,esi	;кол-во найденных троек
27.	.m1:	
28.	cmp ebx,0	
29.	jne .m2	
30.	bt eax,edx	
31.	jc .skip	
32.	inc ebx	
33.	jmp .skip	
34.	.m2:	
35.	cmp ebx,1	
36.	jne .m3	
37.	bt eax,edx	
38.	jnc .skip	
39.	inc ebx	
40.	jmp .skip	
41.	.m3:	
42.	bt eax,edx	
43.	jc .m31	
44.	mov ebx,1	
45.	jmp .skip	
46.	.m31:	
47.	inc esi	
48.	xor ebx,ebx	
49.	.skip:	
50.	dec edx	
51.	cmp edx,-1	
52.	jne .m1	
53.	mov [count],esi	

```

54.    push str2
55.    call [printf]
56.    push [count]           ;ВЫВОД
57.    push strScanInt
58.    call [printf]
59.
60.    call [getch]           ; чтобы пользователь успел увидеть ответ,
    ожидаем нажатия клавиши
61.    push 0
62.    call [ExitProcess]     ;завершаем работу программы
63.
64.section '.idata' import data readable
65.    library kernel, 'kernel32.dll',\
66.        msvcrt, 'msvcrt.dll',\
67.        user32, 'USER32.DLL'
68.
69.include 'api\user32.inc'
70.include 'api\kernel32.inc'
71.    import kernel,\
72.        ExitProcess, 'ExitProcess',\
73.        HeapCreate, 'HeapCreate',\
74.        HeapAlloc, 'HeapAlloc'
75. include 'api\kernel32.inc'
76.    import msvcrt,\
77.        printf, 'printf',\
78.        scanf, 'scanf',\
79.        getch, '_getch'
80.

```

