

Национальный исследовательский университет
«Высшая школа экономики»

Факультет компьютерных наук

Департамент

Программной инженерии

Самостоятельная работа
по дисциплине
«Архитектура вычислительных систем»

Тема работы: Вариант 8. Определение количества чисел Перрена от 1 до беззнакового машинного слова
--

Выполнил: студент группы БПИ194
Гребенщиков М. М.

тел. +7 (922) 704 5875
e-mail адрес: mmgrebenschikov@edu.hse.ru

Преподаватель: Легалов Александр Иванович

Москва, 2020

Структура работы

1. 8option.asm – содержит исходный код программы
2. 8option.exe – скомпилированный исполняемый файл
3. Note.pdf – Пояснительная записка работы

В коде программы находятся комментарии, описывавшие каждый шаг программы.

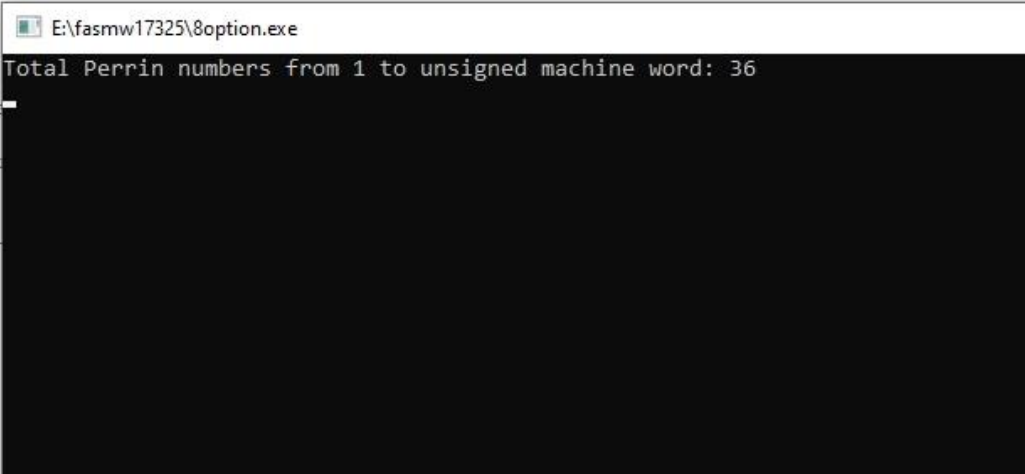
Тестирование программы

Так как программа не принимает входных данных, результат работы был сопоставлен с данными, полученные путём альтернативного подсчёта. Область допустимых входных значений в данном случае не ограничена.

```
table
;сим Михайлович БПИ194-----
;ределяющую число чисел Перрена от 1 до беззнакового машинного слова-----

in numbers from 1 to unsigned machine word: %d', 10, 0

;Первые 3 числа
; 0 - не в диапа
ьное значение ма
cutable
-----
пускаем функции
```



Тест программы

Переменные для промежуточных данных: one, two, three, four

Алгоритм работы программы

1. Объявление всех необходимых переменных
2. Запуск цикла подсчёта количества чисел Перрена
3. Расчёт очередного числа Перрена

4. На каждой итерации цикла происходит сдвиг четырёх ключевых переменных при помощи макроса
5. Проверка подсчитанного числа на принадлежность границам беззнакового машинного слова
6. Корректировка полученных данных ввиду повторов чисел
7. Вывод полученной информации в консоль и завершение работы программ

Список литературы

1. https://ru.wikipedia.org/wiki/%D0%A7%D0%B8%D1%81%D0%BB%D0%BE_%D0%9F%D0%B5%D1%80%D1%80%D0%B5%D0%BD%D0%B0
2. <https://oeis.org/A001608>

Приложение. Код программы

```

1.  format PE console
2.
3.  entry start
4.
5.  include 'win32a.inc'
6.
7.  section '.data' data readable writable
8.
9.  ;-----Вариант 8 Гребенщиков Максим Михайлович БПИ194-----
10. ;-----Разработать программу, определяющую число чисел Перрена от 1 до беззнакового машинного
    слова
11.
12.
13.     formatNum db '%d', 10, 0
14.     outputLine db "Total Perrin numbers from 1 to unsigned machine word: %d", 10, 0
15.
16.     one dd 3    ;Позиция x-3
17.     two dd 0    ;Позиция x-2
18.     three dd 2  ;Позиция x-1 ;Первые 3 числа не учитываем,
19.     four dd ?   ;Позиция x ; 0 - не в диапазоне, а 3 и 2 - повторяются
20.
21.     MAXWORD = 65536 ;Максимальное значение машинного слова + 1, так как в цикле строгое
    сравнение
22.
23. section '.code' code readable executable
24. ;-----
25.     start:
26.         call calcNumbs ;Запускаем функции
27.         call output
28.
29.     finish:
30.         call [getch]
        push NULL

```

```

31.         call [ExitProcess]
32.
33. ;-----
34.
35.     macro shiftt
36.     {
37.         mov edx, [two] ;При каждой итерации сдвигаем наши числа на 1 влево
38.         mov [one], edx
39.         mov edx, [three]
40.         mov [two], edx
41.         mov edx, [four]
42.         mov [three], edx
43.     }
44.
45. ;-----
46.
47.     calcNumbs:
48.         mov ebx, -1 ;Считаем итоговое значение
49.             ;Начальное -1, тк в цикле лишняя итерация
50.     lp:
51.         add ebx, 1
52.         mov eax, [one]
53.         add eax, [two] ;Вычисляем текущее число через сумму (x-2) + (x-3)
54.         mov [four], eax
55.
56.         shiftt ;используем макрос сдвига
57.
58.         cmp [four], MAXWORD ;Сравниваем текущее x с максимальным значение слова
59.         jl lp
60.         add ebx, -1 ;Повтор числа 5 в наборе - вычитаем 1
61.         ret
62.
63. ;-----
64.     output:
65.         push ebx ;Выводи ответ в консоль
66.         push outputLine
67.         call [printf]
68.         jmp finish ;Завершаем выполнение программы
69.
70. ;-----
71.
72. section '.idata' import data readable
73.
74.     library kernel, 'kernel32.dll',\
75.         msvcrt, 'msvcrt.dll'
76.
77.     import kernel,\
78.         ExitProcess, 'ExitProcess'
79.
80.     import msvcrt,\
81.         printf, 'printf',\
82.         getch, '_getch',\
83.         scanf, 'scanf'
84. ;-----

```