



Отчёт по лабораторной работе № 5 по курсу "Фундаментальная информатика"

Студент группы М8О-104Б-22 Чесноков Василий, № по списку 20

Контакты www, e-mail, icq, skype chesnokov_vd@mail.ru

Работа выполнена: « 26 » 09 2022 г.

Преподаватель Потенко М. А. каф. 806

Входной контроль знаний с оценкой

Отчёт сдан « » 201 г., итоговая оценка

Подпись преподавателя

1. Тема: Машина Тьюринга

2. Цель работы: Научиться пользоваться машиной Тьюринга в четвёрках

3. Задание (вариант № 19): Вычисление двоичного циклического сдвига первого числа вправо на число разрядов, равное второму

4. Оборудование(лабораторное):

ЭВМ, процессор, имя узла сети с ОП Мб,

НМД Мб. Терминал адрес. Принтер

Другие устройства

Оборудование ПЭВМ студента, если использовалось:

Процессор Intel Core i5 с ОП 16384 Мб, НМД 524288 Мб. Монитор Retina 13,3"

Другие устройства клавиатура, мышь

5. Программное обеспечение(лабораторное):

Операционная система семейства, наименование версия

интерпретатор команд версия

Система программирования версия

Редактор текстов версия

Утилиты операционной системы

Прикладные системы и программы

Местонахождение и имена файлов программ и данных

Программное обеспечение ЭВМ студента, если использовалось:

Операционная система семейства macOS, наименование macOS Monterey версия 12.3.1

интерпретатор команд zsh версия 5.8

Система программирования версия

Редактор текстов версия

Утилиты операционной системы

Прикладные системы и программы

Местонахождение и имена файлов программ и данных на домашнем компьютере

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

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

```
che4ve@MacBook-Pro-Vasilij mainprog % gcc main.c -g
2022-10-26 15:41:48.748 xcodebuild[926:37157] Requested but did not find extension point with identifier Xcode.IDEKit.ExtensionSentinelHostApplications for extension Xcode.DebuggerFoundation.AppExtensionHosts.watchOS of plug-in com.apple.dt.IDEWatchSupportCore
2022-10-26 15:41:48.741 xcodebuild[926:37157] Requested but did not find extension point with identifier Xcode.IDEKit.ExtensionPointIdentifierToBundleIdentifier for extension Xcode.DebuggerFoundation.AppExtensionToBundleIdentifierMap.watchOS of plug-in com.apple.dt.IDEWatchSupportCore
che4ve@MacBook-Pro-Vasilij mainprog % sudo gdb a.out
Password:
GNU gdb (GDB) 12.1
Copyright (C) 2022 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-apple-darwin21.5.0".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://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...
Reading symbols from /Users/che4ve/MAI/c_codes/mainprog/a.out.dSYM/Contents/Resources/DWARF/a.out...
(gdb)
```

```
(gdb) list 0
1      #include <stdio.h>
2      #include <stdlib.h>
3
4      // Задаем константы
5      #define I_0 0
6      #define J_0 15
7      #define L_0 10
8      #define K_0 50
9
10     // Функция нахождения минимума двух чисел
(gdb)
11     int min(int a, int b) {
12         return (a >= b) ? b : a;
13     }
14
15     // Функция нахождения максимума двух чисел
16     int max(int a, int b) {
17         return (a >= b) ? a : b;
18     }
19
20     // Функция нахождения знака числа
(gdb)
21     short sign(int a) {
22         if (a > 0) return 1;
23         if (a < 0) return -1;
24         return 0;
25     }
26
27     // Определяем структуру - вектор
28     struct vector {
29         int i, j;
30     };
(gdb)
31
32     // Функция, возвращающая объект структуры вектор с переданными точками x, y
33     struct vector set_vector(int x, int y) {
34         struct vector new_vector;
35         new_vector.i = x;
36         new_vector.j = y;
37         return new_vector;
38     }
39
40 (gdb)
```

Пункты 1-7 отчета составляются **строго до** начала лабораторной работы.

Допущен к выполнению работы. Подпись преподавателя _____

8. Распечатка протокола (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

```
(gdb) list cross_product
45     struct vector C;
46
47     };
48
49     // Функция нахождения произведения векторов
50     int cross_product(struct vector A, struct vector B) {
51         return A.i * B.j - B.i * A.j; // Возвращает координату z полученного вектора
52     }
53
```

9. **Дневник отладки** должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) 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. **Замечания автора** по существу работы Размер кода машины Тьюринга для исполнения поставленной задачи составил 226 строчек. В протоколе не удалось разместить его полностью, поэтому он прилагается в дополнительном файле.

11. Выводы

Я научился пользоваться машиной Тьюринга с форматом команд "в четверках". Работа с ней помогла мне лучше понять принцип реализации алгоритмов и процессов пошагового вычисления.

Недочёты при выполнении задания могут быть устранены следующим образом:

Подпись студента В. Тескоков