

# Отчет по лабораторной работе № 25 по курсу “Фундаментальная информатика”

Студент группы М8О-103Б-21

Батулин Евгений Андреевич, № по списку 2

Контакты: e-mail: uggin@inbox.ru, telegram: @uggin0

Работа выполнена: «14» мая 2021 г.

Преподаватель: каф. 806 Севастьянов Виктор Сергеевич

Отчет сдан «    » \_\_\_\_\_ 20\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

1. **Тема:** Автоматизация сборки программ модульной структуры на языке Си с использованием утилиты make
2. **Цель работы:** изучить принципы работы make
3. **Задание:** составить makefile для модульной программы из ЛР26, протестировать результаты работы, запротоколировать его
4. **Оборудование** (студента):  
Процессор *Intel Core i9-9980HK ES(QQLS)*, 8c/16t @ 4.4GHz с ОП 32768 Мб, НМД 6656 Гб. Монитор 1920x1080
5. **Программное обеспечение** (студента):  
Операционная система семейства: *Windows*, наименование: *10*, версия *1809 LTSC*  
интерпретатор команд: *MSYS (c MinGW)* версия *1.3.0.0*.  
Система программирования -- версия --, редактор текстов *Visual Studio Code*, версия *1.66.2*  
Утилиты операционной системы:  
Прикладные системы и программы: *gcc*, *notepad++*
6. **Идея, метод, алгоритм** решения задачи

Исходниками программы являются *main.c*, *list.c* и *list.h*. Для компиляции объектных файлов и их компоновки необходимо выбрать компилятор и компоновщик соответственно. В моём *makefile*, как и в примере, используется *gcc*. Для отладки программы следует оставить стандартную отладочную информацию при помощи флага *-g*, что и было сделано при помощи «*CCFLAGS = -g..*». При помощи «*main.o: list.h*» я обозначил зависимость *main* от *list.h*. После этого необходимо скомпилировать объектные файлы *main.o* и *list.o*, что и было сделано при помощи «*main.o: main.c list.h / CC -c main.c*» и «*list.o: list.c list.h / CC -c list.c*». Сборка конечной программы производится при помощи команды «*CC -o main main.o list.o*».

## 7. Сценарий выполнения работы

1. Запуск текстового редактора
2. Создание *make*
3. Проверка работоспособности программы после её сборки
4. Отладка параметров сборки / исходников сборки
5. Протоколирование работы *make*
6. Завершение работы

## 8. Распечатка протокола

*makefile*

*CC = gcc*  
*LD = gcc*

*CCFLAGS = -g -Wall -pedantic -std=c99*

*LDFLAGS =*

*main: main.o list.o*  
    *CC -o main main.o list.o*  
*main.o: main.c list.h*  
    *CC -c main.c*

```
list.o: list.c list.h
CC -c list.c
main.o: list.h
```

make

```
Meta@DESKTOP MINGW64 /c/Users/Meta/Desktop/MAI/LABS/25
$ make
CC -c list.c
CC -o main main.o list.o
```

## 9. Дневник отладки

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание
1	дом	14\05\2022	13:14	В сборке не использовалась переменная выбора компилятора	Замена «сс» на «СС» в командах сборки	

## 10. Замечания автора по существу работы

## 11. Выводы

Подводя итог, в процессе данной работы я познакомился с утилитой make и научился автоматической сборке модульных программ.

Подпись студента \_\_\_\_\_