

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тульский государственный университет»

Интернет-институт

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ

по дисциплине

«Операционные системы и среды»

Семестр 2

Вариант 3

Выполнил: студент гр. ИБ262521-ф

Артемов Александр Евгеньевич

Проверил: канд. техн. наук, доц.

Французова Юлия Вячеславовна

Тула 2023

Задание.

На основе командного файла TASM.BAT разработать командный файл MASM.BAT, предназначенный для автоматизации процесса трансляции программы на языке Microsoft Assembler.

Выполнение лабораторной работы.

1. Запуск трансляции при помощи командного файла TASM.BAT.

Согласно пункта 2.11 Основных теоретических сведений исходный код командного файла TASM.BAT представлен ниже:

```
@ECHO OFF
REM Командный файл автоматизации компиляции программы REM на языке Ассемблера.
Используется драйвер ANSY.SYS CLS
REM Проверить наличие параметров в командной строке
IF (%1)==( ) ECHO Esc[31;1mВ командной строке не задан параметр!Esc[0m IF (%1)==( )
GOTO END
REM Проверка существования файлов
IF NOT EXIST %1.ASM ECHO Esc[31;1mНеверное имя файла!Esc[0m
IF NOT EXIST %1.ASM GOTO END
REM Задать цвет символов и фона для вывода заставки ECHO Esc[37;44m
ECHO *****
ECHO Система программирования Turbo Assembler Version 4.0
ECHO Программист – студент гр. 620292 Иванов С.П.
ECHO *****
REM Вернуться к нормальному изображению
ECHO Esc[0m
REM Запомнить координаты курсора
ECHO Esc[s
:LOOP
REM Восстановить координаты курсора
ECHO Esc[u
ECHO Транслируется файл %1.ASM. Создается файл листинга %1.LST TASM /L /T /ZI
/N %1,NUL,%1
IF ERRORLEVEL 1 ECHO Esc[31;1mОшибка трансляции!Esc[0m
IF ERRORLEVEL 1 GOTO END
CHOICE /C:YN /N:T:N,8 Создать объектный модуль (Да – Y, Нет – N):
IF ERRORLEVEL 2 GOTO END
IF ERRORLEVEL 1 ECHO Второй этап трансляции. Создается файл %1.OBJ IF ERRORLEVEL
1 TASM %1,%1,NUL
CHOICE /C:ECN /N /T:N,8 Создать EXE-файл – E, COM-файл – C, отказ – N:
IF ERRORLEVEL 3 GOTO END
IF ERRORLEVEL 2 GOTO ONE
IF ERRORLEVEL 1 GOTO TWO
:ONE
TLINK /T %1.OBJ
GOTO END
:TWO
TLINK %1.OBJ
:END
REM Сдвинуть список параметров влево
SHIFT
REM Переход на метку LOOP, если параметры еще доступны
```

```
IF NOT "%1"==" " GOTO LOOP
ECHO Конец работы командного файла
```

Для запуска данного командного файла использована операционная система Windows 7 32-bit. Borland Turbo Assembler взят со стены VK (https://vk.com/topic-205404099_47749107).

После пары часов попыток завести ассемблер, командный файл приобрел следующий вид:

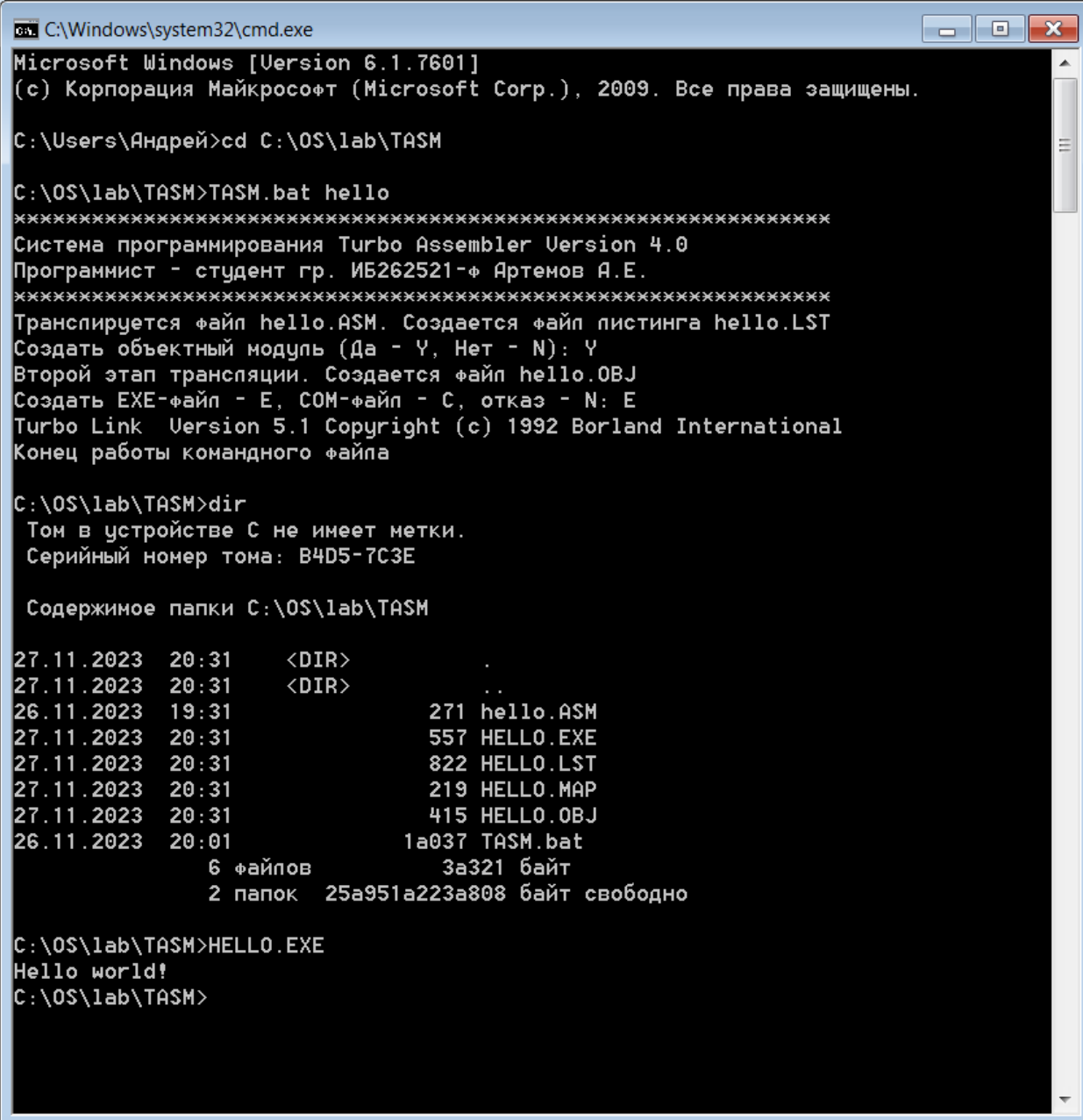
```
@ECHO OFF
IF (%1)==( ) ECHO В командной строке не задан параметр IF (%1)==( ) GOTO END
IF NOT EXIST %1.ASM ECHO Неверное имя файла!
IF NOT EXIST %1.ASM GOTO END
ECHO *****
ECHO Система программирования Turbo Assembler Version 4.0
ECHO Программист - студент гр. ИБ262521-ф Артемов А.Е.
ECHO *****
ECHO Транслируется файл %1.ASM. Создается файл листинга %1.LST
C:\TASM\TASM /L /T /ZI /N %1
IF ERRORLEVEL 1 ECHO Ошибка трансляции!
IF ERRORLEVEL 1 GOTO END
CHOICE /C:YN /N /T 8 /D Y /M "Создать объектный модуль (Да - Y, Нет - N):"
IF ERRORLEVEL 2 GOTO END
IF ERRORLEVEL 1 ECHO Второй этап трансляции. Создается файл %1.OBJ
CHOICE /C:esn /N /T 8 /D e /M "Создать EXE-файл - E, COM-файл - C, отказ - N:"
IF ERRORLEVEL 3 GOTO END
IF ERRORLEVEL 2 GOTO ONE
IF ERRORLEVEL 1 GOTO TWO
:ONE
TLINK /T %1.OBJ
GOTO END
:TWO
TLINK %1.OBJ
:END
SHIFT
IF NOT "%1"==" " GOTO LOOP
ECHO Конец работы командного файла
```

Как видно, из кода были удалены комментарии, Esc-последовательности, исправлены параметры команды CHOICE. Для отладки командного файла был использован файл hello.ASM приведенный ниже: (<https://becooladmin.wordpress.com/2020/10/12/hello-world-on-tasm/>)

```
assume CS:code,DS:data
code segment
begin:
mov AX,data
mov DS,AX
mov AH, 09h
mov DX, offset msg
int 21h
mov AX, 4C00h
int 21h
code ends
data segment
msg db "Hello world!$"
data ends
```

```
stk segment stack
db 256 dup (?)
stk ends
end begin
```

Такой код на ассемблере ничего не делает, кроме вывода сообщения «Hello world!». Процесс трансляции и линковки программы hello.exe при помощи командного файла TASM.bat представлен на следующем снимке окна командной строки:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(с) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\Андрей>cd C:\OS\lab\TASM

C:\OS\lab\TASM>TASM.bat hello
*****
Система программирования Turbo Assembler Version 4.0
Программист - студент гр. ИБ262521-Ф Артемов А.Е.
*****
Транспируется файл hello.ASM. Создается файл листинга hello.LST
Создать объектный модуль (Да - Y, Нет - N): Y
Второй этап трансляции. Создается файл hello.OBJ
Создать EXE-файл - E, COM-файл - C, отказ - N: E
Turbo Link Version 5.1 Copyright (с) 1992 Borland International
Конец работы командного файла

C:\OS\lab\TASM>dir
Том в устройстве C не имеет метки.
Серийный номер тома: B4D5-7C3E

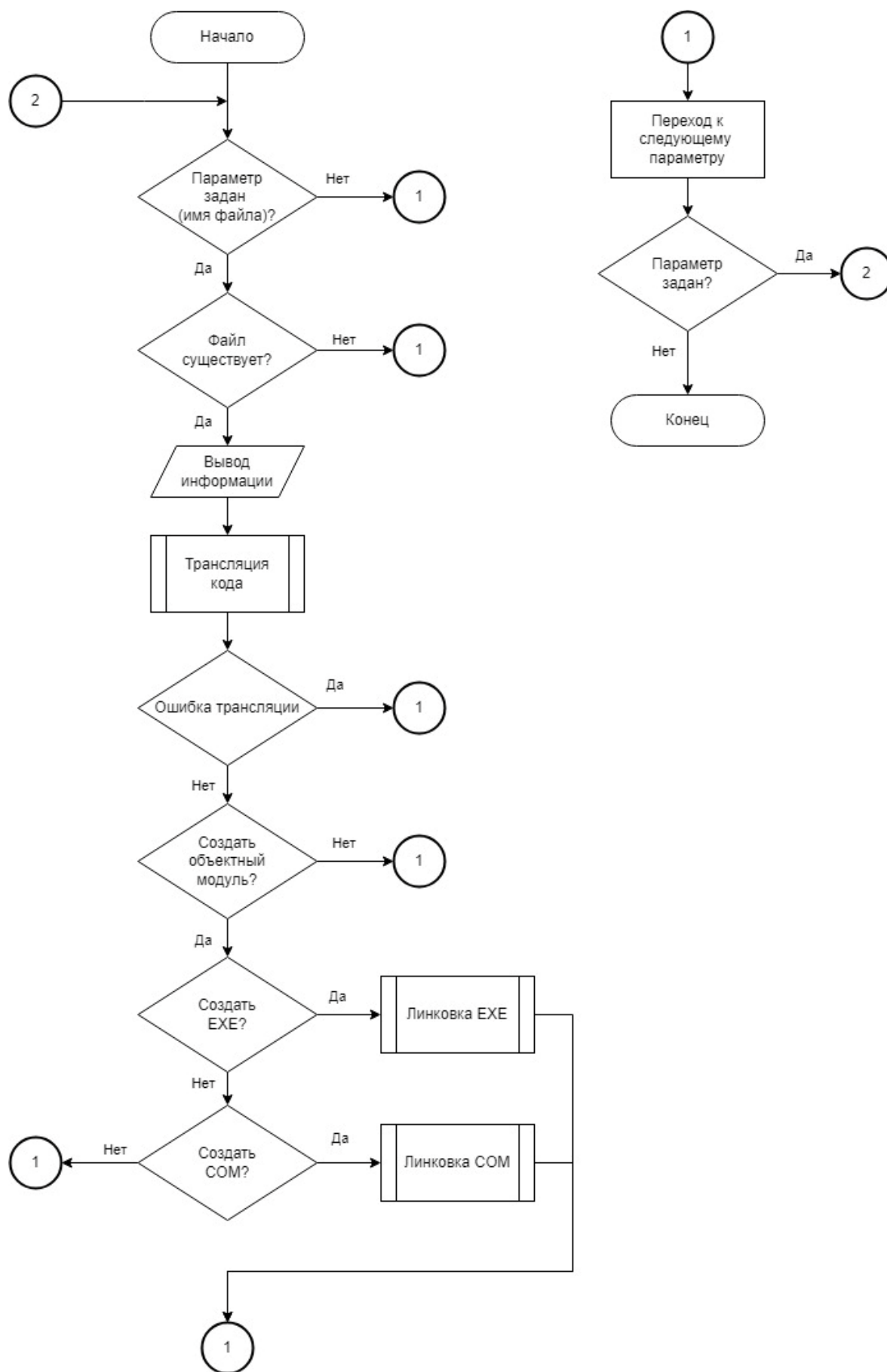
Содержимое папки C:\OS\lab\TASM

27.11.2023  20:31    <DIR>          .
27.11.2023  20:31    <DIR>          ..
26.11.2023  19:31                271 hello.ASM
27.11.2023  20:31                557 HELLO.EXE
27.11.2023  20:31                822 HELLO.LST
27.11.2023  20:31                219 HELLO.MAP
27.11.2023  20:31                415 HELLO.OBJ
26.11.2023  20:01            1a037 TASM.bat
                6 файлов                3a321 байт
                2 папок  25a951a223a808 байт свободно

C:\OS\lab\TASM>HELLO.EXE
Hello world!
C:\OS\lab\TASM>
```

Как видно из вывода программы, кроме исполняемого файла создаются объектный файл, листинг и маппинг. Исполняемый файл программы работает как положено. Перейдем к созданию командного файла MASM.BAT, использующий Microsoft Assembler.

2. Блок – схема алгоритма работы командного файла TASM.BAT



Алгоритм работы командного файла TASM.BAT:

- проверяется наличие входного параметра;
- проверяется существование файла исходного кода, переданного в параметре;
- выводится информация о студенте и компиляторе;
- трансляция исходного кода в файл – листинг кода ассемблера Borland Turbo Assembler;
- при ошибке трансляции переход на обработку следующего входного параметра;
- запрос пользователю на создание объектного файла, при отказе переход на обработку следующего входного параметра;
- запрос пользователю на выбор формата исполнительного файла (EXE или COM), при отказе переход на обработку следующего входного параметра;
- линковка объектного файла в исполнительный файл (при сборке COM – файла используется флаг \t);
- переход (команда SHIFT) к следующему входному параметру и его проверка, при наличии параметра переход в начало алгоритма;
- при отсутствии следующего входного параметра окончание работы командного файла.

3. Запуск трансляции при помощи командного файла MASM.BAT.

Microsoft Assembler взят по адресу: <http://masm32.masmcode.com/masm32/masm32v11r.zip>.

Исходный код командного файла MASM.BAT:

```
@ECHO OFF
IF (%1)==() ECHO В командной строке не задан параметр IF (%1)==() GOTO END

IF NOT EXIST %1.ASM ECHO Неверное имя файла!
IF NOT EXIST %1.ASM GOTO END
ECHO *****
ECHO Система программирования Macro Assembler Version 6.14.844
ECHO Программист - студент гр. ИБ262521-ф Артемов А.Е.
ECHO *****
ECHO Транслируется файл %1.ASM.
C:\masm32\bin\ml.exe /c /coff %1.ASM
IF ERRORLEVEL 1 ECHO Ошибка трансляции!
IF ERRORLEVEL 1 GOTO END
CHOICE /C:YN /N /T 8 /D Y /M "Создать объектный модуль (Да - Y, Нет - N):"
IF ERRORLEVEL 2 GOTO END
IF ERRORLEVEL 1 ECHO Второй этап трансляции. Создается файл %1.OBJ
CHOICE /C:en /N /M "Создать EXE-файл - E, отказ - N:"
IF ERRORLEVEL 2 GOTO END
IF ERRORLEVEL 1 GOTO ONE
:ONE
C:\masm32\bin\link /SUBSYSTEM:CONSOLE /LIBPATH:c:\masm32\lib %1.OBJ
GOTO END
:END
SHIFT
IF NOT "%1"==" " GOTO LOOP
ECHO Конец работы командного файла
```

Для отладки командного файла был использован файл hello.ASM приведенный ниже:

(http://www.i2r.ru/static/378/out_16816.shtml)

```
.386
.model flat,stdcall
option casemap:none

includelib kernel32.lib

SetConsoleTitleA PROTO :DWORD
GetStdHandle PROTO :DWORD
WriteConsoleA PROTO :DWORD,:DWORD,:DWORD,:DWORD,:DWORD
ExitProcess PROTO :DWORD
Sleep PROTO :DWORD

.const
sWriteText db 'Hello world!!!!'

.code
Main PROC
```

```
LOCAL hStdout :DWORD

push -11
call GetStdHandle
mov hStdout,EAX

push 0
push 0
push 16d
push offset sWriteText
push hStdout
call WriteConsoleA

push 2000d
call Sleep

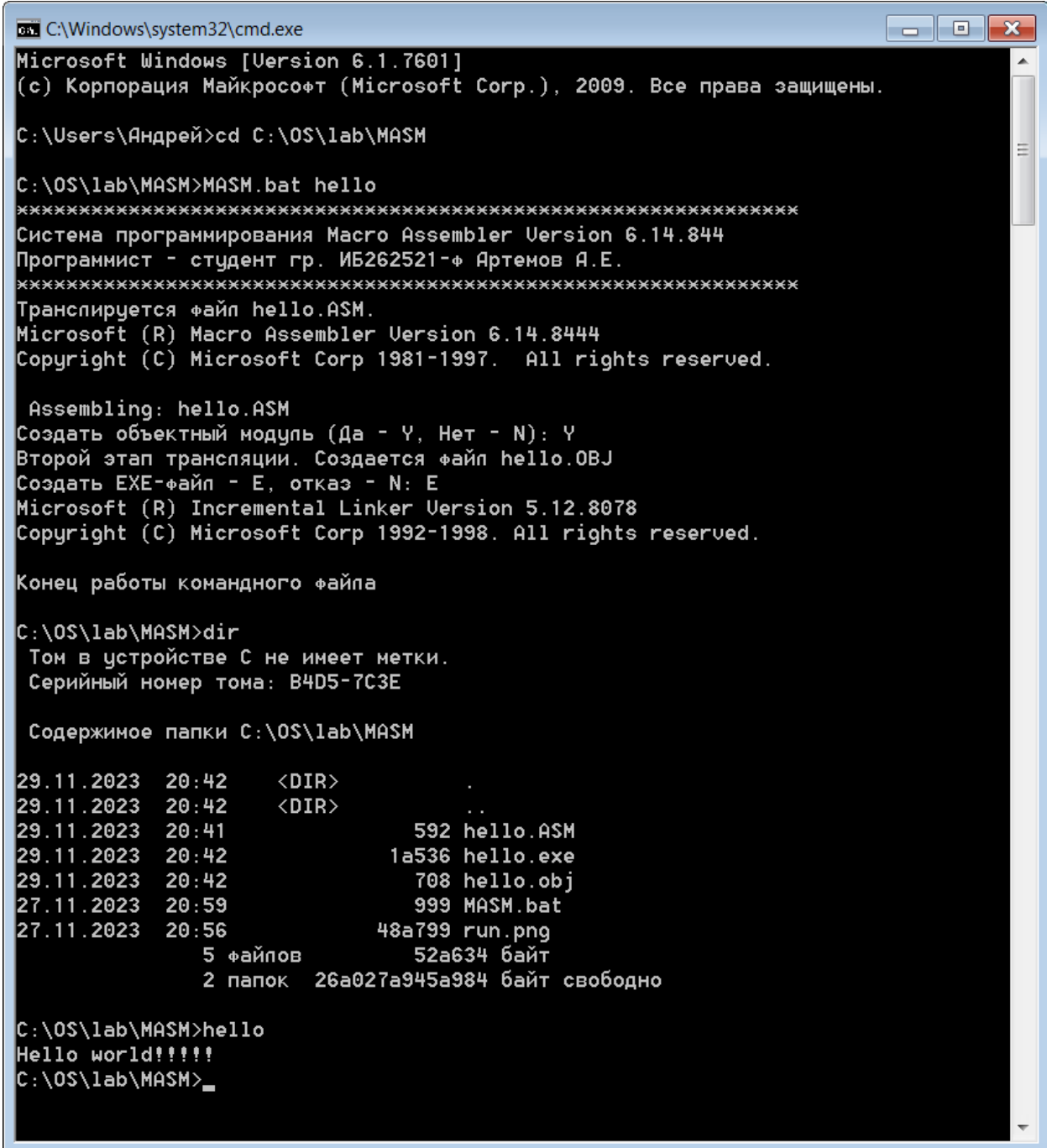
push 0
call ExitProcess

Main ENDP

end Main
```

Данный код выводит сообщение «Hello world!!!!!» с задержкой 2 секунды после.

Процесс трансляции и линковки программы hello.exe при помощи командного файла MASM.bat представлен на следующем снимке окна командной строки:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\Андрей>cd C:\OS\lab\MASM

C:\OS\lab\MASM>MASM.bat hello
*****
Система программирования Macro Assembler Version 6.14.844
Программист - студент гр. ИБ262521-Ф Артемов А.Е.
*****
Транспируется файл hello.ASM.
Microsoft (R) Macro Assembler Version 6.14.8444
Copyright (C) Microsoft Corp 1981-1997. All rights reserved.

Assembling: hello.ASM
Создать объектный модуль (Да - Y, Нет - N): Y
Второй этап трансляции. Создается файл hello.OBJ
Создать EXE-файл - E, отказ - N: E
Microsoft (R) Incremental Linker Version 5.12.8078
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.

Конец работы командного файла

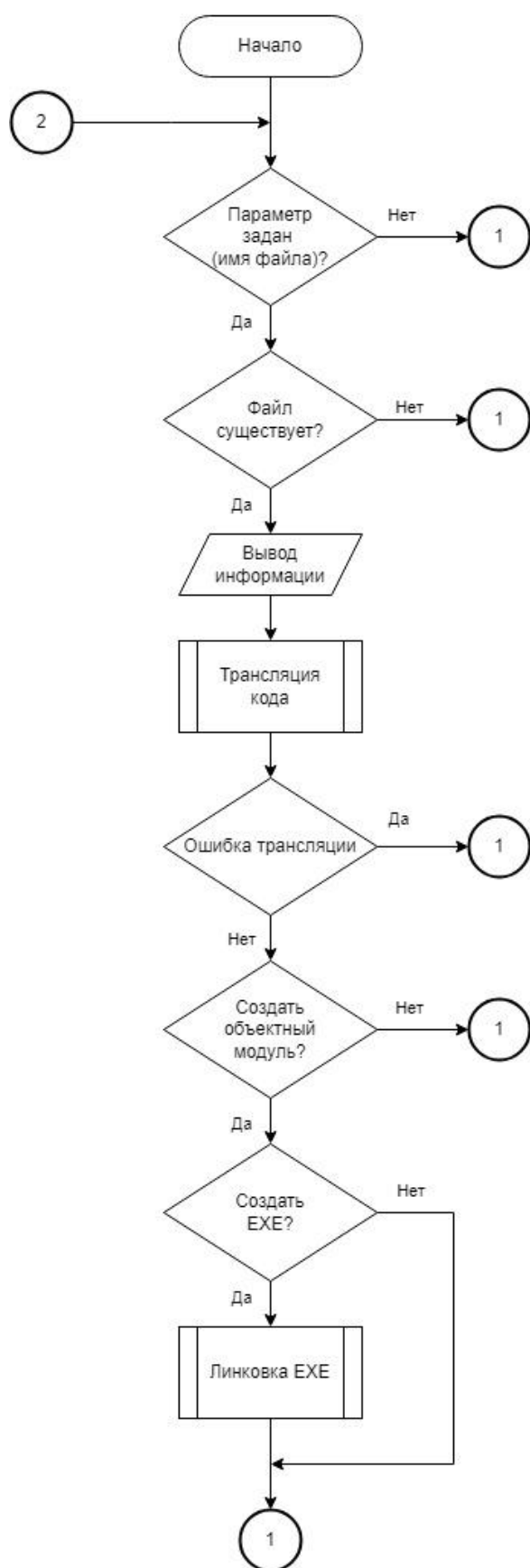
C:\OS\lab\MASM>dir
Том в устройстве C не имеет метки.
Серийный номер тома: B4D5-7C3E

Содержимое папки C:\OS\lab\MASM

29.11.2023  20:42    <DIR>          .
29.11.2023  20:42    <DIR>          ..
29.11.2023  20:41                592 hello.ASM
29.11.2023  20:42             1a536 hello.exe
29.11.2023  20:42                708 hello.obj
27.11.2023  20:59                999 MASM.bat
27.11.2023  20:56             48a799 run.png
                5 файлов             52a634 байт
                2 папок   26a027a945a984 байт свободно

C:\OS\lab\MASM>hello
Hello world!!!!
C:\OS\lab\MASM>
```

4. Блок – схема алгоритма работы командного файла MASM.BAT



Алгоритм работы командного файла MASM.BAT:

- проверяется наличие входного параметра;
- проверяется существование файла исходного кода, переданного в параметре;
- выводится информация о студенте и компиляторе;
- запрос пользователю на создание объектного файла, при отказе переход на обработку следующего входного параметра ();
- трансляция исходного кода в объектный файл (флаг /с – сборка объектного файла, флаг /coff задает формат файлов объектов для сборки на Win32);
- при ошибке трансляции переход на обработку следующего входного параметра;
- запрос пользователю на создание исполнительного файла (EXE), при отказе переход на обработку следующего входного параметра;
- линковка объектного файла в исполнительный файл (флаг /SUBSYSTEM:CONSOLE – для сборки консольного приложения, /LIBPATH:c:\masm32\lib – задает путь к библиотекам Macro Assembler);
- переход (команда SHIFT) к следующему входному параметру и его проверка, при наличии параметра переход в начало алгоритма;
- при отсутствии следующего входного параметра окончание работы командного файла.

Ответы на контрольные вопросы.

1. Каково назначение командных файлов в операционной системе ПК?

С помощью командных файлов пользователь ПК может разработать собственные оригинальные средства общения с вычислительной техникой, отсутствующие среди стандартных возможностей ОС.

2. Каким образом можно создать командный файл?

При помощи любого текстового редактора.

3. Какими способами в командный файл можно ввести исходные данные?

Перечислить при запуске в терминале после названия командного файла.

4. Каким образом можно принудительно прервать выполнение командного файла?

Комбинацией клавиш Ctrl+C.

5. Можно ли в командном файле программировать звуковые сигналы?

Да, командой BEEP [/Dn] [/Fm] [/Rp] [/Wq] или BEEP файл [/E].

6. Можно ли из командного файла управлять позицией вывода текстовой строки на дисплей?

Да, при помощи ANSI – драйвера и Esc-последовательностей.

7. Что такое формальные и фактические параметры командного файла? Сколько формальных и фактических параметров может иметь командный файл?

Фактические параметры передаются командному файлу при запуске в терминале. Формальные параметры используются в коде командного файла в формате %0, %1, %2 и так далее, где %0 – имя самого командного файла, а %1, %2 – это 1-ый и 2-ой переданные параметры. Количество передаваемых параметров от 1 до 9.

8. Какие функции выполняют в операционной системе переменные окружения?

Основное предназначение переменных окружения — конфигурирование системы и программ.

9. Каким образом осуществляется создание, модификация и уничтожение переменных окружения?

Установка значений переменных окружения производится с помощью команд PATH, PROMPT, SET.

10. Каково назначение операторов перенаправления ввода-вывода?

Перенаправление стандартных потоков в определённое пользователем место, например, в файл, а также периферийные устройства посимвольного типа (AUX, CON, COM1 – COM4, LPT1 – LPT3, NUL, PRN).

11. Каково назначение конвейеров данных?

Перенаправить вывод данных одной программы или команды ОС на вход другой программы или команды.

12. Какие функции выполняют команды-фильтры?

Преобразование потока данных по определённым правилам.

13. Какие способы отладки могут быть применены к командным файлам?

Использование необязательного переключателя /DEBUG, который служит для отладки командных файлов, обеспечивая отображение кода возврата на экране монитора.

14. Как в командном файле программировать вывод информации в Куда?

15. Можно ли в командном файле программировать вывод информации на печатающее устройство?

Да, перенаправлением потока вывода команды ECHO.

16. Можно ли в командном файле программировать вывод информации файл на магнитном диске?

Да, перенаправлением потока вывода команды ECHO в файл.

17. Каким образом в алгоритме работы командного файла программируются разветвления?

При помощи команды IF.

18. Можно ли с помощью языка командных файлов программировать циклические алгоритмы?

При помощи команд FOR и LOOP.

19. Каким образом из командного файла можно управлять цветом

выводимых на дисплей символов?

При помощи команд COLOR.

20. Каким образом в командном файле запрограммировать временную Временную что?

21. Каким образом в командном файле можно проверить существования файла на магнитном диске?

При помощи команды IF EXIST [имя файла].

22. Можно ли из командного файла инициировать выполнение другого командного файла?

При помощи команды CALL.

23. Каким образом можно отобразить на экране содержимое текстового файла?

При помощи команд TYPE или MORE.

24. Что такое код завершения программы? Каким образом можно использовать код завершения программы в командном файле?

Число от 0 до 16, где 0 означает успешное выполнение команды, а остальные – код ошибки. Код завершения команды используется в команде IF ERRORLEVEL [число] [команда], где ERRORLEVEL принимает значение истина если код возврата команды не меньше указанного числа.

25. Каким образом в командных файлах программируются циклические алгоритмы?

При помощи команд FOR и LOOP.

26. Для каких целей предназначен драйвер ANSI.SYS?

Он предоставляет пользователю дополнительные возможности управления дисплеем и клавиатурой, которые могут быть использованы и в командных файлах.

27. Будет ли работать командный файл TASM.BAT (см. пример), если в вычислительной системе отсутствует ANSI-драйвер?

Будет, но Esc-последовательности будет выводиться как текст.

28. Каково назначение программы Norton Batch Enhancer?

Для расширения возможностей командных файлов.

29. Можно ли в командном файле запрограммировать перезапуск операционной системы?

При помощи команды SHUTDOWN /r.

30. Какие изменения и дополнения введены в технологию командных файлов в операционных системах Windows NT/2000/XP?

осуществлять ряд специальных функций управления, для реализации которых рекомендуется применять пакетную обработку с помощью командных файлов. Таким типовыми задачами управления являются, например,

- конфигурирование, остановка и рестарт сервисов;
- управление обновлениями операционной системы;
- управление системным реестром;
- управление списками управления доступом (ACL);

- настройка приложений;
- генерация отчетов о состоянии системы;
- управление файлами и их разделением;
- добавление, изменение и удаление профилей пользователей.

Обработку команд, вводимых пользователем из командной строки операционной системы Windows NT/2000/XP, а также выполнение командных файлов осуществляет интерпретатор команд (командный процессор) CMD.EXE. Интерпретатор CMD.EXE поддерживает работу с несколькими новыми командами (например, AT, COLOR, START, SETLOCAL, ENDLOCAL и др.) и имеет возможность включать или выключать режим расширенной обработки командных файлов. Если режим расширенной обработки выключен, то командные файлы, разработанные по правилам для операционных систем MS-DOS и Windows 95/98/Me, будут выполнены. При расширенной обработке команд изменения и/или добавления затрагивают следующие команды: DEL или ERASE, COLOR, CD или CHDIR, MD или MKDIR, PROMPT, PUSH, POP, SET, SETLOCAL, ENDLOCAL, IF, FOR, CALL, SHIFT, GOTO, START, ASSOC, FTYPE.

31. Возможны в командных файлах арифметические операции над числовыми значениями аргументов?

При помощи команды SET с параметром /A.