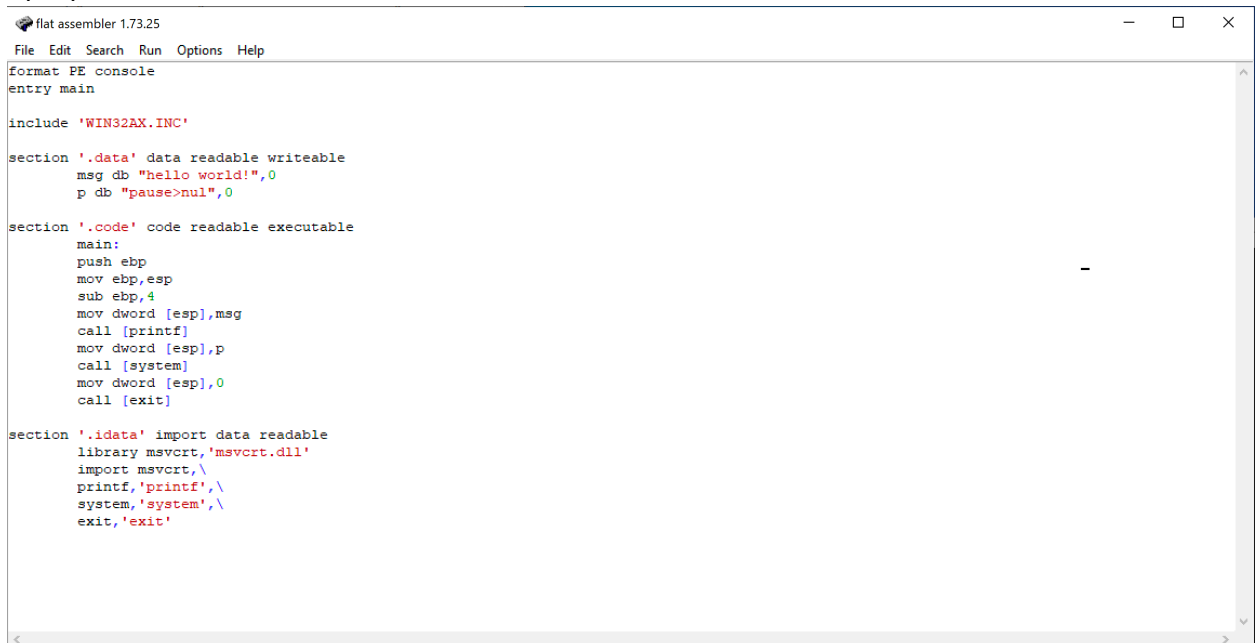


# Домашняя работа по первому семинару

(Студент: Лапшин Даниил Павлович БПИ198)

## Программа №1:

Программа выводит в консоль "hello world!"



```
format PE console
entry main

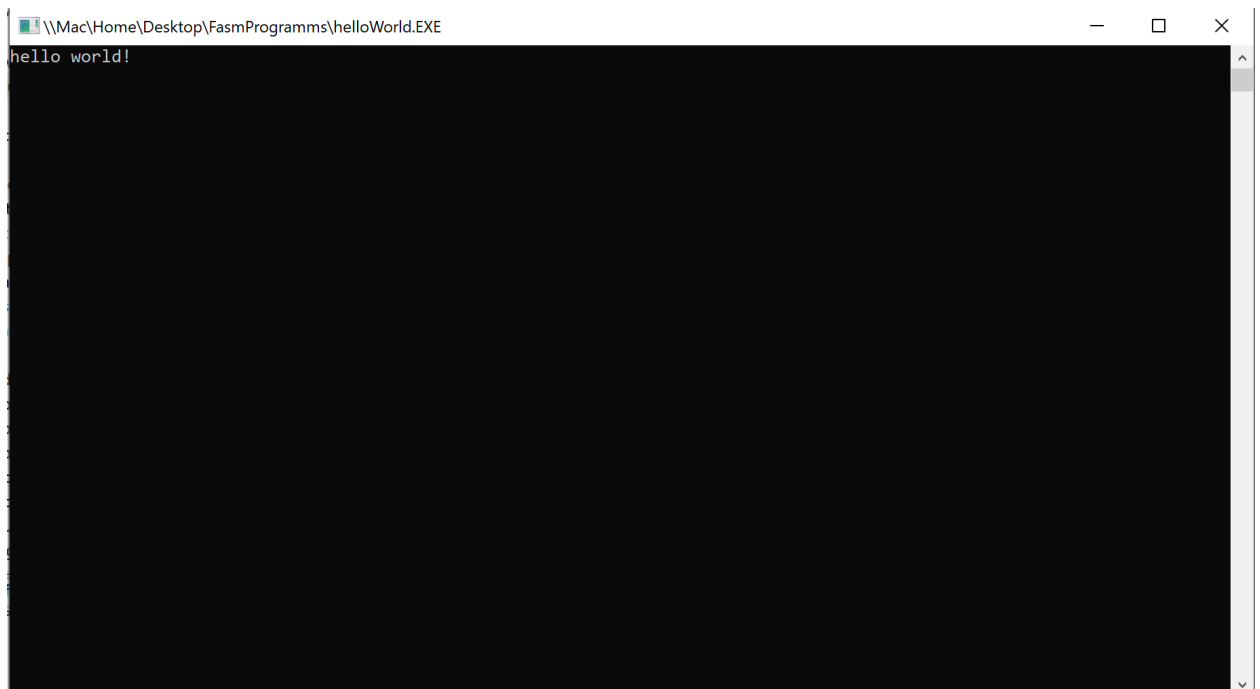
include 'WIN32AX.INC'

section '.data' data readable writeable
    msg db "hello world!",0
    p db "pause>nul",0

section '.code' code readable executable
main:
    push ebp
    mov ebp,esp
    sub ebp,4
    mov dword [esp],msg
    call [printf]
    mov dword [esp],p
    call [system]
    mov dword [esp],0
    call [exit]

section '.idata' import data readable
    library msvcrt,'msvcrt.dll'
    import msvcrt,\
        printf,'printf',\
        system,'system',\
        exit,'exit'
```

Результат выполнения программы:



```
\\Mac\Home\Desktop\FasmProgramms\helloWorld.EXE
hello world!
```

## Программа №2:

Программа выводит содержимое заданного текстового файла на экран



```
format pe console 4.0
include 'WIN32AX.INC'

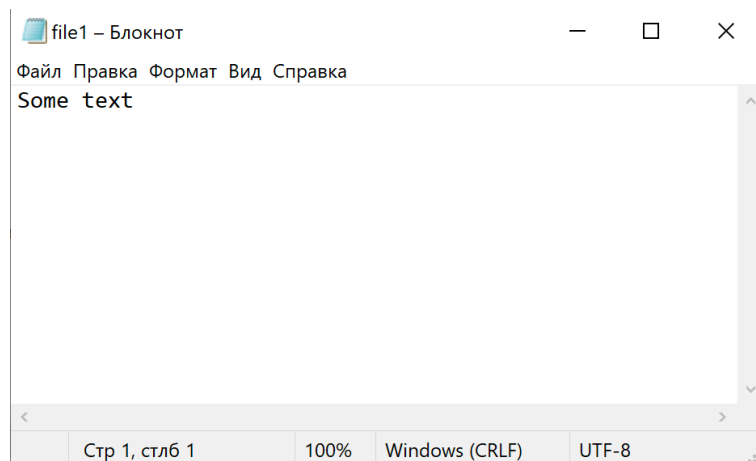
.data
    FileName db 'file1.txt',0
    hFile dd ?
    nSize dd ?
    lpBytesRead dd ?
    lpBuffer rb 8192

    MessageBoxCaption db 'Output:',0

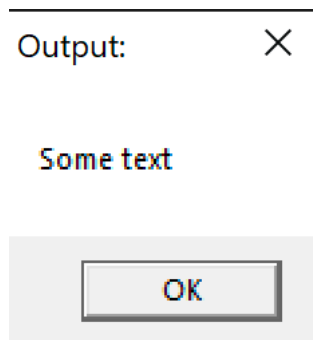
.code
main:
    invoke CreateFile, FileName, GENERIC_READ, 0, 0, OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL, 0 ; Open the file (to get its handle)
    ; TODO: TestApiError
    mov [hFile], eax ; Save the file's handle to hFile
    invoke GetFileSize, [hFile], 0 ; Determine the file size
    ; TODO: TestApiError
    mov [nSize], eax ; Save the file size given by EAX
    invoke ReadFile, [hFile], lpBuffer, [nSize], lpBytesRead, 0 ; Now read the full file
    ; TODO: TestApiError
    invoke CloseHandle, [hFile] ; Handle should be closed after the file has been read
    invoke MessageBox, NULL, addr lpBuffer, addr MessageBoxCaption, MB_OK ; Easy way of outputting the text
    invoke ExitProcess, 0

.end main
```

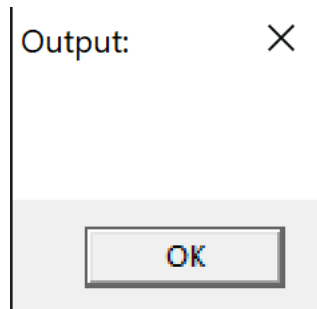
## Содержимое файла:



## Результат выполнения программы:



Если неправильно задать имя файла, то программа выполнится, но в окне будет пустая строка



Пример №3:

Программа считывает из консоли ваше имя и выводит сообщение “Hello <ваше имя>”

```
flat assembler 1.73.25
File Edit Search Run Options Help
format PE console
entry main
include 'macro/import32.inc'

section '.data' data readable writeable
msg db "Enter your name: ",0
chrarr db "%s",0
msg2 db "Hello %s",0dh,0ah,0
p db "pause",0

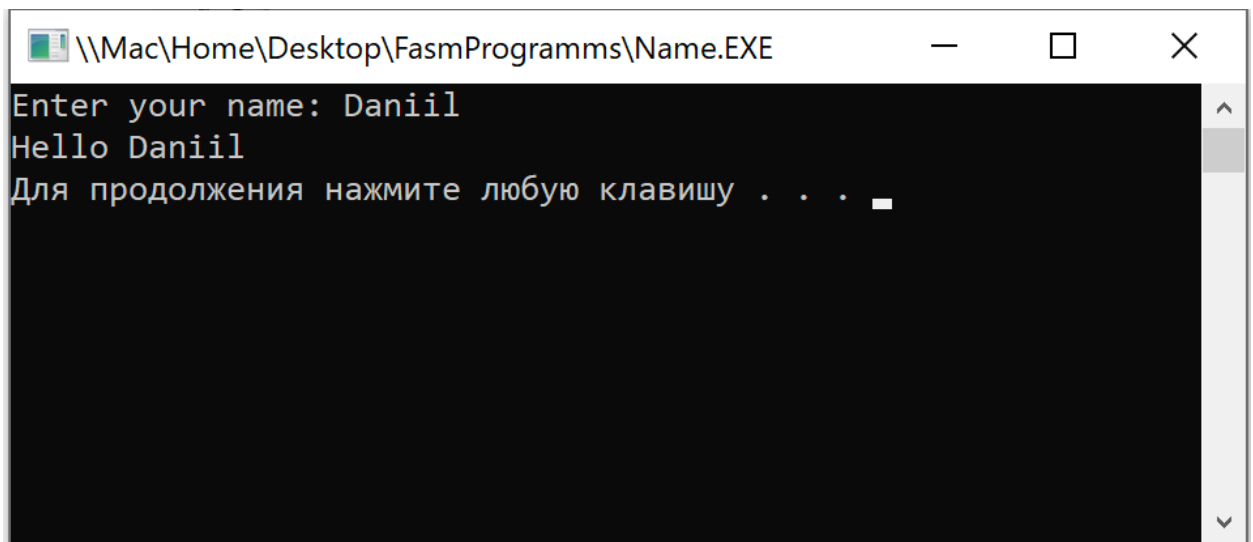
section '.code' code readable executable

main:
push ebp
mov ebp,esp
sub esp,24
mov dword [esp],msg
call [printf]
lea eax,[ebp-12]
mov dword [esp+4],eax
mov dword [esp],chrarr
call [scanf]
mov dword [esp],msg2
call [printf]
mov dword [esp],p
call [system]
mov dword [esp],0
call [exit]

section '.idata' import data readable
library msvcrt,'msvcrt.dll'

import msvcrt,\
printf,'printf',\
scanf,'scanf',\
system,'system',\
exit,'exit'
```

Результат выполнения программы:



Пример №4:

Программа выводит в консоль числа от 1 до 10 с использованием цикла

```
flat assembler 1.73.25
File Edit Search Run Options Help
format PE console
entry main

include 'macro/import32.inc'

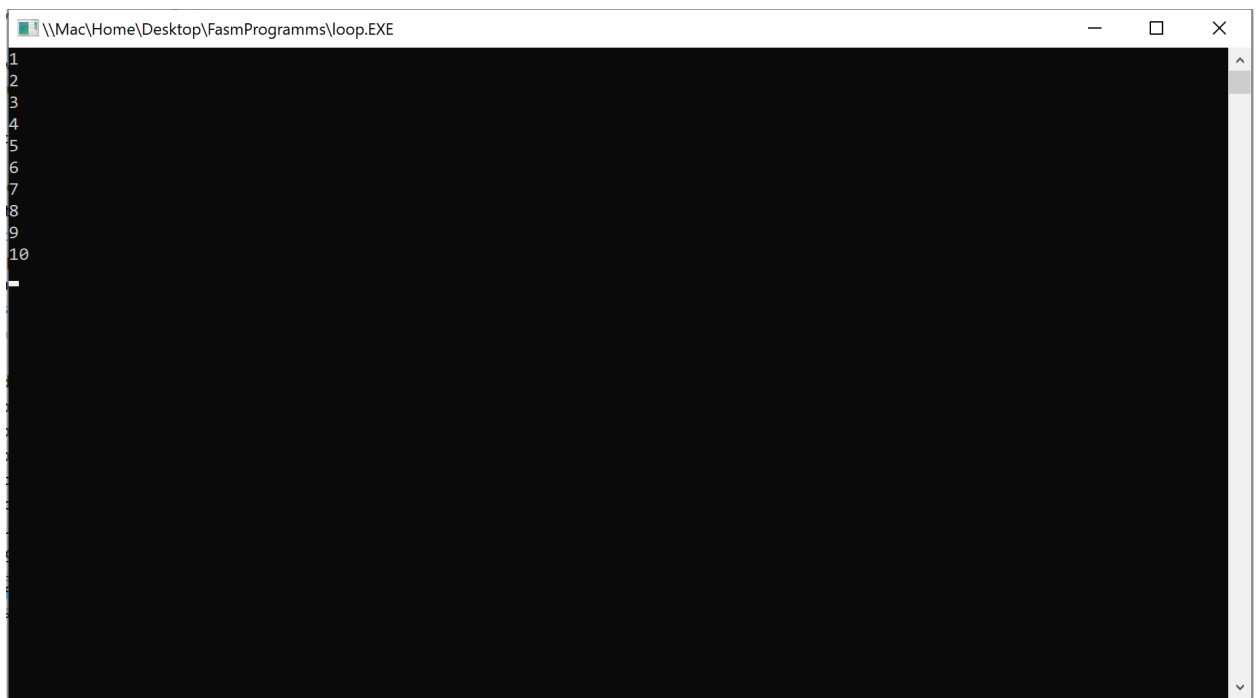
section '.data' data readable writeable
    int1 db "%d",0dh,0ah,0
    p db "pause>nul",0

section '.code' code readable executable
    main:
        push ebp
        mov ebp,esp
        sub esp,12
        mov dword [ebp-4],1
    loopA:
        mov eax,dword [ebp-4]
        cmp eax,11
        jge BreakA
        mov eax,dword [ebp-4]
        mov dword [esp+4],eax
        mov dword [esp],int1
        call [printf]
        mov eax,dword [ebp-4]
        inc eax
        mov dword [ebp-4],eax
        jmp loopA
    BreakA:
        add esp,4
        mov dword [esp],p
        call [system]
        mov dword [esp],0
        call [exit]

section '.idata' import data readable
    library msvcrt,'msvcrt.dll'

    import msvcrt,\
        printf,'printf',\
        scanf,'scanf',\
        system,'system',\
        exit,'exit'
```

Результат выполнения программы:



### Пример №5:

Программа скачивает fasm.zip с официального сайта и сохраняет его в текущую папку

```

flat assembler 1.73.25
File Edit Search Run Options Help
;=====
format PE_CONSOLE 4.0
entry start

include 'win32ax.inc'
include 'macro\if.inc'

INTERNET_OPEN_TYPE_PRECONFIG = 0

;-----
macro println arg*
{
    cinvoke printf, '%s', arg
}
;-----

section '.data' data readable writeable
    wsadata      WSADATA
    TestApp      db "InetURL/1.0",0
    strURL       db "http://flatassembler.net/fasmg.zip",0
    zipfile      db "fasmg.zip",0 ; download to executable's folder
    cannotopenfile db "Cannot download!",0
    errorzip     db "Unexpected Error!",0
    zipmode      db "wb",0
    downloadmsg  db "==== Download Successful! ====",0
    DownloadTitle db "Status Report",0

;-----
section '.code' code readable executable
start:
    invoke WSASStartup, 0002h, wsadata
    println 'Trying to download fasmg.zip file...'
    stdcall httpdownloadfile
    invoke WSACleanup

.finished:
    invoke ExitProcess,0

;-----
proc httpdownloadfile
local hInet:DWORD, hFile:DWORD, htmfip:DWORD, dwRead:DWORD, buffer[1024]:BYTE
    invoke InternetOpen,TestApp,INTERNET_OPEN_TYPE_PRECONFIG,NULL,NULL,0
    mov [hInet] eax

```

flat assembler 1.73.25
File Edit Search Run Options Help

```

cinvoke fclose,[htmlfp]
ret
endp

;-----
section '.idata' import data readable writeable
library kernel32, 'kernel32.dll',\
user32, 'user32.dll',\
comctl32, 'comctl32.dll',\
shell32, 'shell32.dll',\
advapi32, 'advapi32.dll',\
comdlg32, 'comdlg32.dll',\
gdi32, 'gdi32.dll',\
wsock32, 'wsock32.dll',\
wininet, 'wininet.dll',\
msvcrt, 'msvcrt.dll'

include 'api\kernel32.inc'
include 'api\user32.inc'
include 'api\comctl32.inc'
include 'api\shell32.inc'
include 'api\advapi32.inc'
include 'api\comdlg32.inc'
include 'api\gdi32.inc'
include 'api\wsock32.inc'

import wininet,\
InternetOpen, 'InternetOpenA',\
InternetOpenUrl, 'InternetOpenUrlA',\
InternetReadFile, 'InternetReadFile',\
InternetCloseHandle, 'InternetCloseHandle'

import msvcrt,\
printf, 'printf',\
fopen, 'fopen',\
fwrite, 'fwrite',\
fclose, 'fclose',\
fprintf, 'fprintf',\
fgets, 'fgets'

; end of file =====

```

flat assembler 1.73.25
File Edit Search Run Options Help

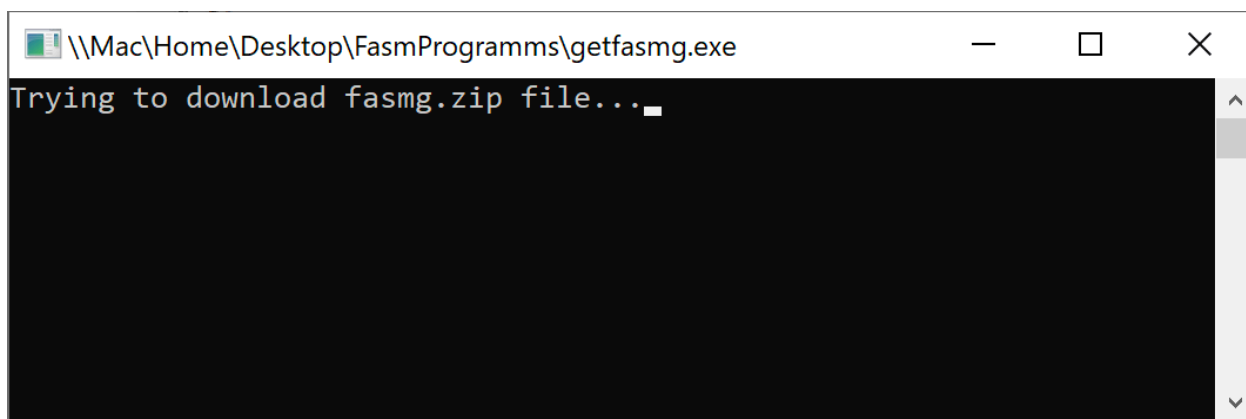
```

;-----
proc httpdownloadfile
local hINet:DWORD, hFile:DWORD, htmlfp:DWORD, dwRead:DWORD, buffer[1024]:BYTE
invoke InternetOpen,TestApp,INTERNET_OPEN_TYPE_PRECONFIG,NULL,NULL,0
mov [hINet], eax
invoke InternetOpenUrl,[hINet],strURL,NULL,0, 0, 0
mov [hFile], eax
.if [hFile] <> 0
cinvoke fopen,zipfile,zipmode
mov [htmlfp], eax
.if [htmlfp] = NULL
pushad
invoke MessageBox, NULL,cannotopenfile,errorzip,MB_SYSTEMMODAL+MB_OK+MB_ICONWARNING
popad
jmp .finish
.endif
.repeat
invoke InternetReadFile,[hFile],addr buffer,1024,addr dwRead
.if [dwRead] <> 0
cinvoke fwrite,addr buffer,1024,1,[htmlfp]
.endif
.until [dwRead] = 0
invoke MessageBox,NULL,downloadmsg,DownloadTitle,MB_SYSTEMMODAL+MB_OK
.else
invoke InternetCloseHandle,[hINet]
.endif
.finish:
cinvoke fclose,[htmlfp]
ret
endp

;-----
section '.idata' import data readable writeable
library kernel32, 'kernel32.dll',\
user32, 'user32.dll',\
comctl32, 'comctl32.dll',\
shell32, 'shell32.dll',\
advapi32, 'advapi32.dll',\
comdlg32, 'comdlg32.dll',\
gdi32, 'gdi32.dll',\
wsock32, 'wsock32.dll',\
wininet, 'wininet.dll',\
msvcrt, 'msvcrt.dll'

```

Результат выполнения программы:



Когда файл скачается, на экран будет выведено надлежащие сообщение

