

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ «ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Департамент программной инженерии

**Консольное приложение, которое на основе заданной
ASCII-строки символов, решает вопрос, является ли
данная строка палиндромом**

Пояснительная записка

Исполнитель:

Студент группы БПИ198

Мелехин Денис Антонович

31.10.2020

Москва 2020

Текст задания

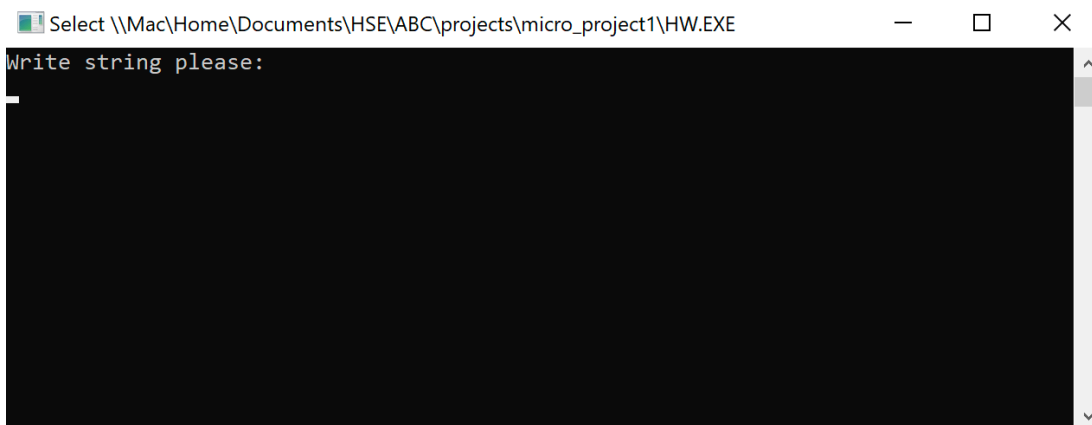
Разработать консольное приложение, которое на основе заданной ASCII-строки символов, решает вопрос, является ли данная строка палиндромом

Описание решения

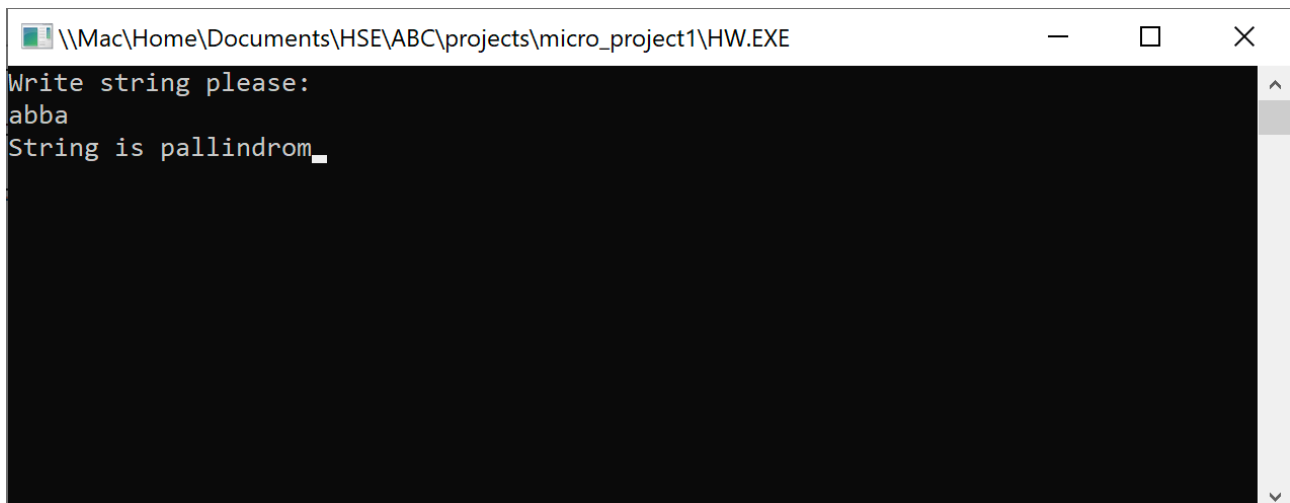
Пользователя просят ввести строку, после чего введенная строка записывается в переменную `input_string`. Далее алгоритм проходится по строке двумя указателями с начала и с конца и проверяет на равенство введенные символы под соответствующими указателями. Если символы не равно - то данная строка не палиндром. Если же все символы прошли проверки на равенство и указатель, который шел с начала превышает указатель, который шел с конца, то введенная строка является палиндромом. Пользователю выводится ответ и алгоритм завершается. Для запуска программы необходимо открыть файл `HW.exe`. Сама программа находится в `HW.ASM`

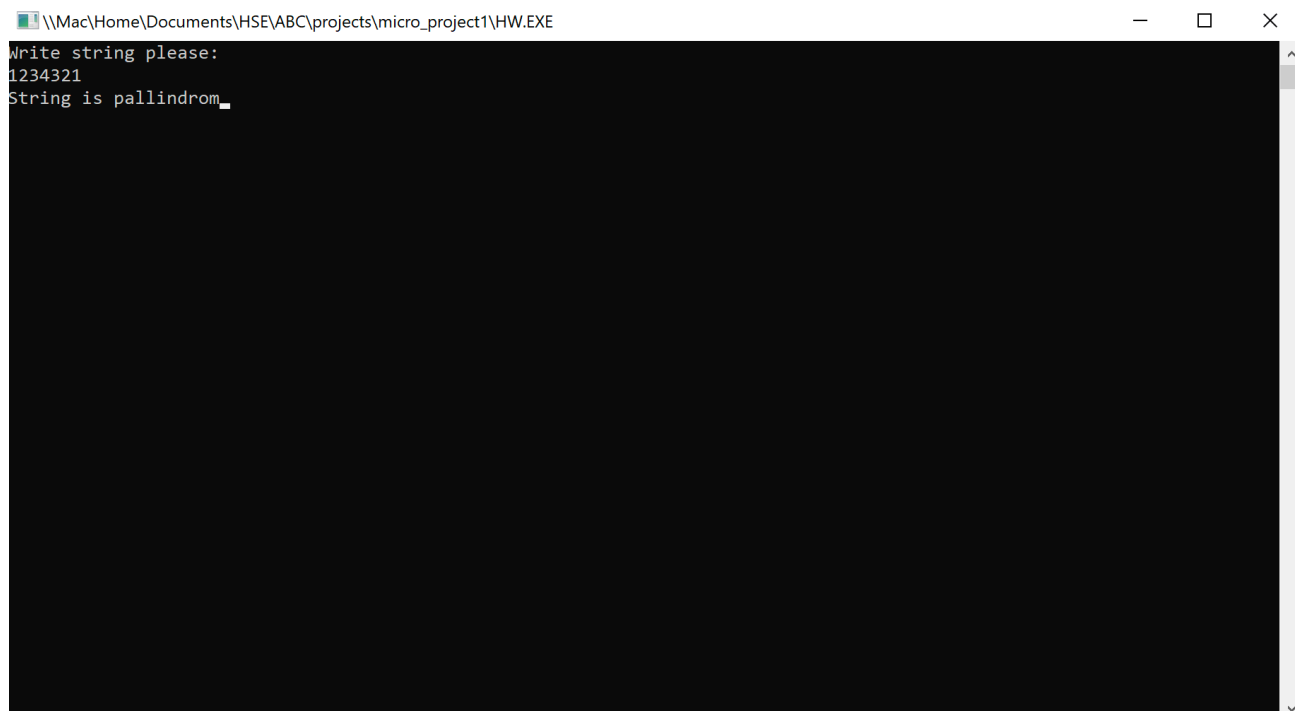
Тестирование программы

Запрос ввода данных:



Тест 1

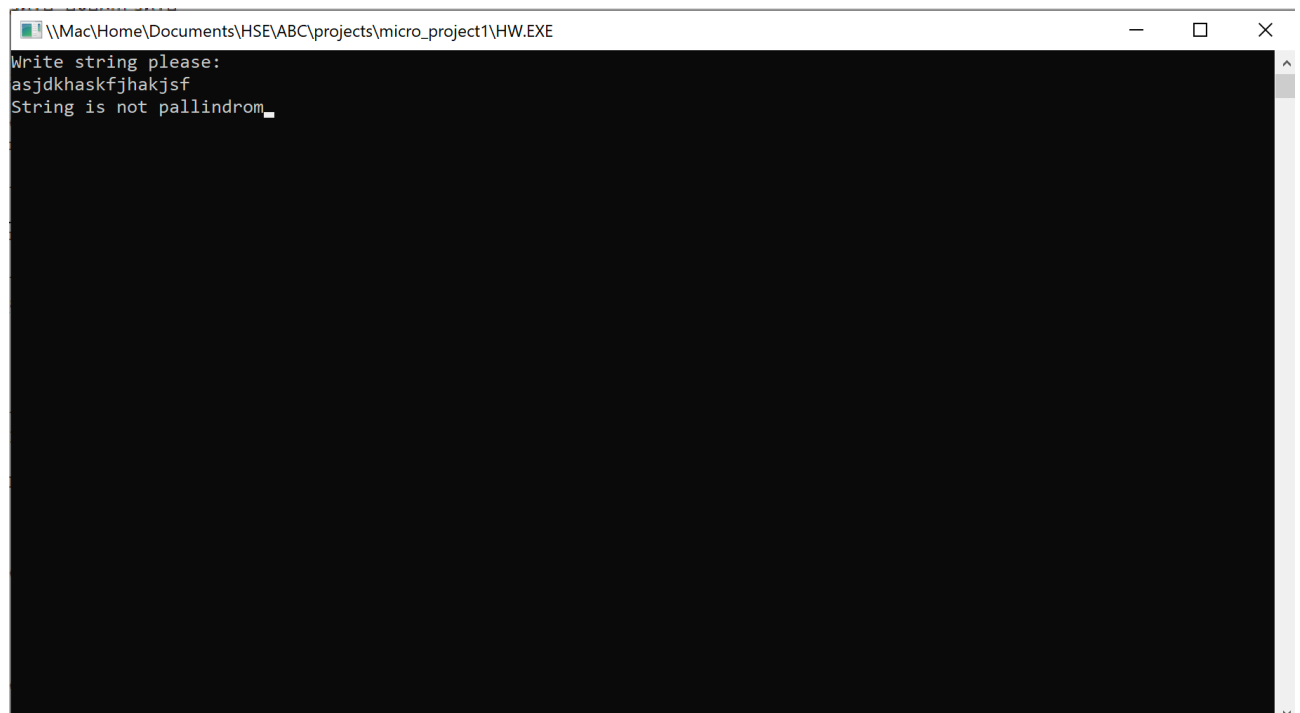




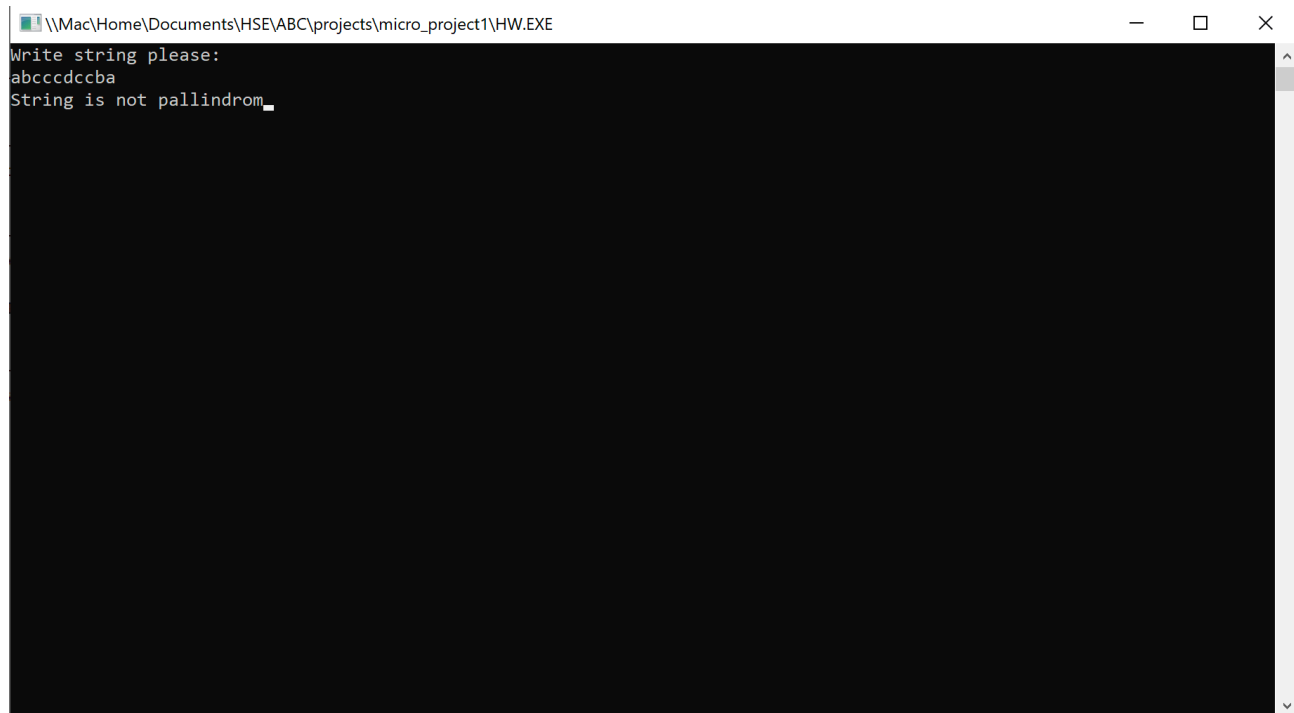
```
\\Mac\\Home\\Documents\\HSE\\ABC\\projects\\micro_project1\\HW.EXE
Write string please:
1234321
String is pallindrom_
```

Тест 2

Тест 3



```
\\Mac\\Home\\Documents\\HSE\\ABC\\projects\\micro_project1\\HW.EXE
Write string please:
asjdkhaskfjhakjsf
String is not pallindrom_
```

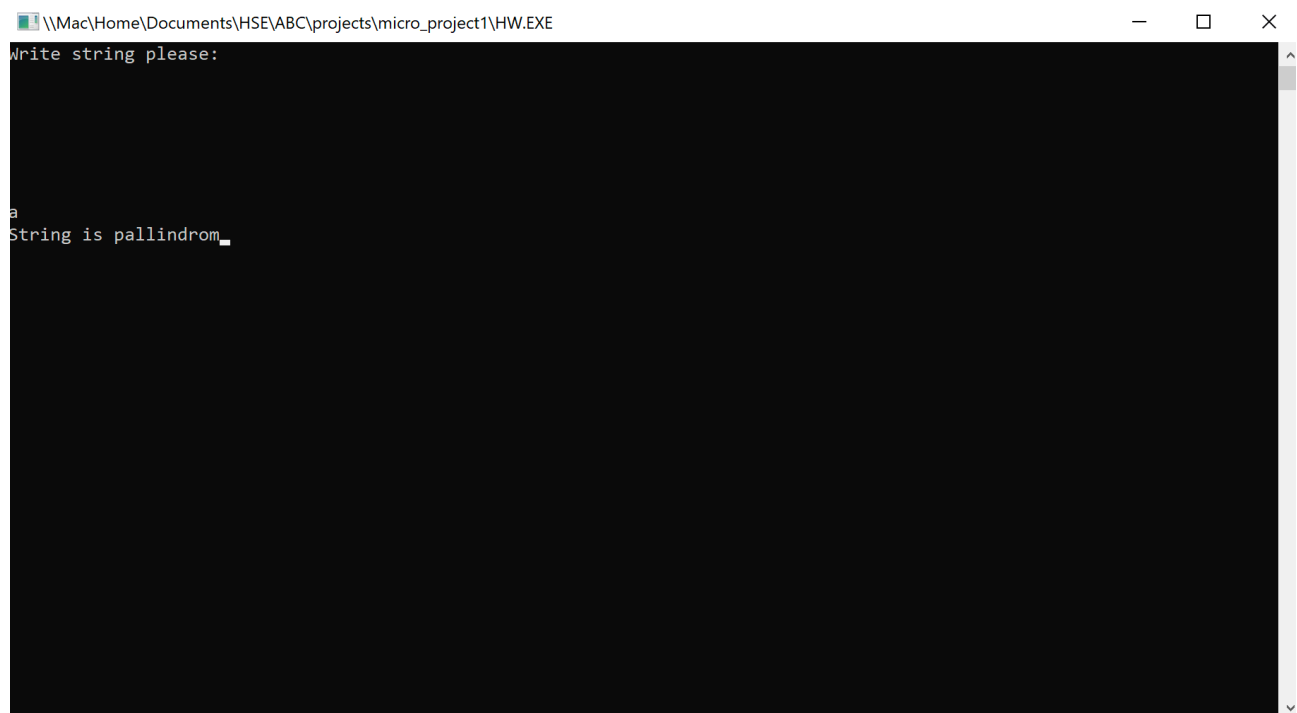


```
\\Mac\Home\Documents\HSE\ABC\projects\micro_project1\HW.EXE
Write string please:
abcccdccba
String is not pallindrom_
```

A screenshot of a terminal window with a black background and white text. The window title bar shows the file path: \\Mac\Home\Documents\HSE\ABC\projects\micro_project1\HW.EXE. The terminal displays the prompt 'Write string please:', followed by the input 'abcccdccba'. Below this, it outputs 'String is not pallindrom_'. The cursor is at the end of the output line.

Тест 4

Тест 5 (при нажатии на enter программа все равно ждет хотя бы один знак)



```
\\Mac\Home\Documents\HSE\ABC\projects\micro_project1\HW.EXE
Write string please:
a
String is pallindrom_
```

A screenshot of a terminal window with a black background and white text. The window title bar shows the file path: \\Mac\Home\Documents\HSE\ABC\projects\micro_project1\HW.EXE. The terminal displays the prompt 'Write string please:', followed by the input 'a'. Below this, it outputs 'String is pallindrom_'. The cursor is at the end of the output line.

Список литературы

1. <http://softcraft.ru/>
2. https://ru.wikipedia.org/wiki/Ассемблер_fasm
3. <http://flatassembler.narod.ru/fasm.htm>

Текст программы

format PE console

include 'win32a.inc'

entry start

section '.data' data readable writable

write_string db 'Write string please:', 10, 0 ; Запрос считывания строки

input_string rd 100 ; Здесь будет храниться введенная строка

only_str db '%s', 0

str_size dd ? ; Размер строки

only_d db '%d', 0

succes db 'String is pallindrom', 0 ; Вывод при введении палиндрома

notSucces db 'String is not pallindrom', 0 ; Вывод при введении не
палиндрома

tmp dd ? ; Переменная для временного хранения регистров

NULL = 0

section '.code' code readable executable

start:

 input_again:

 ; Запрос пользователю

 push write_string

 call [printf]

 ; Считывание строки

 push input_string

 push only_str

 call [scanf]

 ; Считаем размер строки

 push input_string

 call cnt_size

 ; Проверяем на палиндром и выводим ответ

 push input_string

 call check_pallindrom

 ; Завершение алгоритма

 jmp finish

cnt_size:

 ; Не портим регистры

```
push eax
mov eax, esp
push ecx
push edx
push ebx
```

```
xor ecx, ecx
mov edx, [ss:eax+8+0]
```

```
cnt_size_loop:
```

```
mov ebx, 0
```

```
cmp [edx], ebx
je cnt_size_end
```

```
inc ecx
add edx, 1
jmp cnt_size_loop
```

```
cnt_size_end:
```

```
sub eax, 12
mov esp, eax
```

```
mov [str_size], ecx
```

```
pop ebx
pop edx
pop ecx
pop eax
```

```
ret
```

check_pallindrom:

```
push eax
mov eax, esp
mov [tmp], eax
push ecx
push edx
push ebx
```

```
mov ecx, [str_size]
mov edx, [ss:eax+8+0]
mov ebx, [ss:eax+8+0]
```

```
add ebx, ecx
sub ebx, 1
xor ecx, ecx
```

check_pallindrom_loop:

```
cmp edx, ebx
jge check_pallindrom_end_true
```

; Проверка соответствующих символов на равенство

```
mov cl, [edx]
mov al, [ebx]
cmp al, cl
jne check_pallindrom_end_false
```

```
add edx, 1
sub ebx, 1
inc ecx
jmp check_pallindrom_loop
```

; Введенная строка палиндром!

check_pallindrom_end_true:

push succes

call [printf]

jmp finish_pallindrom

; Введенная строка не палиндром!

check_pallindrom_end_false:

push notSucces

call [printf]

finish_pallindrom:

mov eax, [tmp]

sub eax, 12

mov esp, eax

pop ebx

pop edx

pop ecx

pop eax

ret

finish:

call [getch]

push NULL

call ExitProcess

section '.idata' data readable import

```
library kernel, 'kernel32.dll',\  
        msvcrt, 'msvcrt.dll'
```

```
import kernel,\  
        ExitProcess, 'ExitProcess'
```

```
import msvcrt,\  
        printf, 'printf',\  
        scanf, 'scanf',\  
        getch, '_getch'
```