**Завдання 5**

У тексті виділити всі слова, що починаються літерою "к".

**Алгоритм програми**

ВиВиведення слів що починаються на «К»

Пошук у рядку слів що починаються на «к»

Введення рядку символів

**Лістинг програми**

#include "stdafx.h"

#include <string.h>

#include <stdlib.h>

#include <ctype.h>

char \*nextWord(char \*str){

char \*p;

for ( p = str; \*p != '\0' && \*p!=' '; p++ )

;

if ( \*p == '\0' )

return NULL;

while ( \*p==' ')

p++;

return ( \*p == '\0' ) ? NULL : p;

}

char \*getWord(const char \*str, char \*wrd, unsigned int len){

char \*p;

char \*w;

for ( p = (char \*)str, w = wrd; \*p != '\0' && \*p!=' ' ; p++, w++ )

\*w = \*p;

\*w = '\0';

return wrd;

}

int main(){

char buf[500];

char wrd[500];

char \*\*wrds, \*pBuf;

unsigned int count, len, i;

char c;

printf("Enter some string: ");

gets(buf);

printf("Enter some symbol: ");

scanf("%c", &c);

for ( pBuf = buf, count = 0, wrds = NULL; pBuf != NULL; pBuf = nextWord(pBuf) ){

if ( \*pBuf == c ){

getWord(pBuf, wrd, 500);

count++;

if ( count == 1 ){

if ( (wrds = (char\*\*)malloc(sizeof(char\*))) == NULL ){

printf("Memory error!\n");

exit(1);

}

}

else{

if ( (wrds = (char\*\*)realloc(wrds, count \* sizeof(char\*))) == NULL ){

printf("Memory error!\n");

exit(1);

}

}

len = strlen(wrd);

if ( (wrds[count - 1] = (char\*)malloc(len + 1)) == NULL ){

printf("Memory error!\n");

exit(1);

}

strcpy(wrds[count - 1], wrd);

}

}

if ( !count ){

printf("No words starting with '%c' found in string \"%s\"\n", c, buf);

}

/\* и вывод \*/

for ( i = 0; i < count; i++ )

printf("%s\n", wrds[i]);

getchar();

getchar();

return 0;

}

**Результати роботи програми**

