**Question / Answer Flashcards**

Realized by Olga Berezyuk (group А18, Vinnytsia IT-Academy)

**CONTENTS**

1. System requirements

2. About

3. User guide

4. Programmer instruction

5. About author

1. **System requirements**

Operating system Windows XP or higher

CPU Pentium processor at 90 MHz or higher

Memory 5 MB RAM

HD 5 MB available on the hard disk

1. **About**

This program was created to help learn English words. It has two options for

presenting flashcards: EN/RU and RU/EN.

Learning words’ process consists of the making choice of correct translation variant among available varieties. If some word was translated correctly for three times it disappeared from the list, that means a word is learned by user. If any word was translated wrong a correct variant will be displayed for user.

The program shows a statistic list at the end of the flashcards’ cycle.

1. **User guide**

Launch the program.

Choose a menu item:

1. RU/EN

2. EN/RU

Read a shown word and choose one of available variants of translation (1-4). Depending on user’s choice the program displays “Correct!” or “Wrong!!!”.

To add new words open eng.txt file and paste new English words in it one by one and save. Then open rus.txt file and paste translation (words-translation or translation of Eng words) in Russian in it in the same order and save.

To close the program press Esc.

1. **Programmer instruction**

Dev-C ++ is a free integrated development environment. This is an easy IDE that only takes a few minutes to install. This is the best development environment for beginners. Some features of Dev-C ++: lightweight; easy-to-use toolbar; hot keys for compiling and running, for example F9 and F10; simple installation; the ability to create a separate source file.

Structure of project: connecting the necessary libraries, announcement of structure and global variables for work with a list, prototypes of functions and procedures, function main (main menu and call of functions); description of functions and procedures:

Node’s components to storage data about particular word:

typedef struct Words

{

char ru[30]; (Russian word)

char en[30]; (Russian word’s translation into English)

int id; (Node’s id)

int count; (Field-counter of correct user’s answers)

struct Words \*next;

struct Words \*prev;

}\* vocab;

vocab create\_node (creating of node’s template);

void add\_node (procedure adds new word into the list);

vocab RU\_EN\_printout (procedure prints out RU/EN flashcards);

vocab EN\_RU\_printout (procedure prints out EN/RU flashcards);

void fill\_id (procedure fills “id” fields of the list);

void delete\_node (procedure deletes nodes which “count” field value is “3” );

int printout\_your\_choice (procedure process user’s choices);

void finish\_words (procedure prints out a statistic list);

1. **About author**

This project was realized by Olga Berezyuk (group А18, Vinnytsia IT- Academy).