**Національний технічний університет України**

**“Київський політехнічний інститут імені Ігоря Сікорського”**

**Факультет прикладної математики Кафедра системного програмування і спеціалізованих комп’ютерних систем**

**РОЗРАХУНКОВО-ГРАФІЧНА РОБОТА**

* ***дисципліни “Основи програмування”***

**Група: КВ-01**

**Виконала: Рибалка В.С.**

**Оцінка:**

**Київ – 2021**

**Завдання**

Створити ігрову програму мовою програмування С.

Розробка і реалізація ігрових програм має вестися з врахуванням графічних та звукових можливостей, що надаються конкретним комп’ютером.

Програма мусить коректно розв’язувати поставлену задачу. Логічно відокремлені частини алгоритма реалізувати за допомогою окремих функцій.

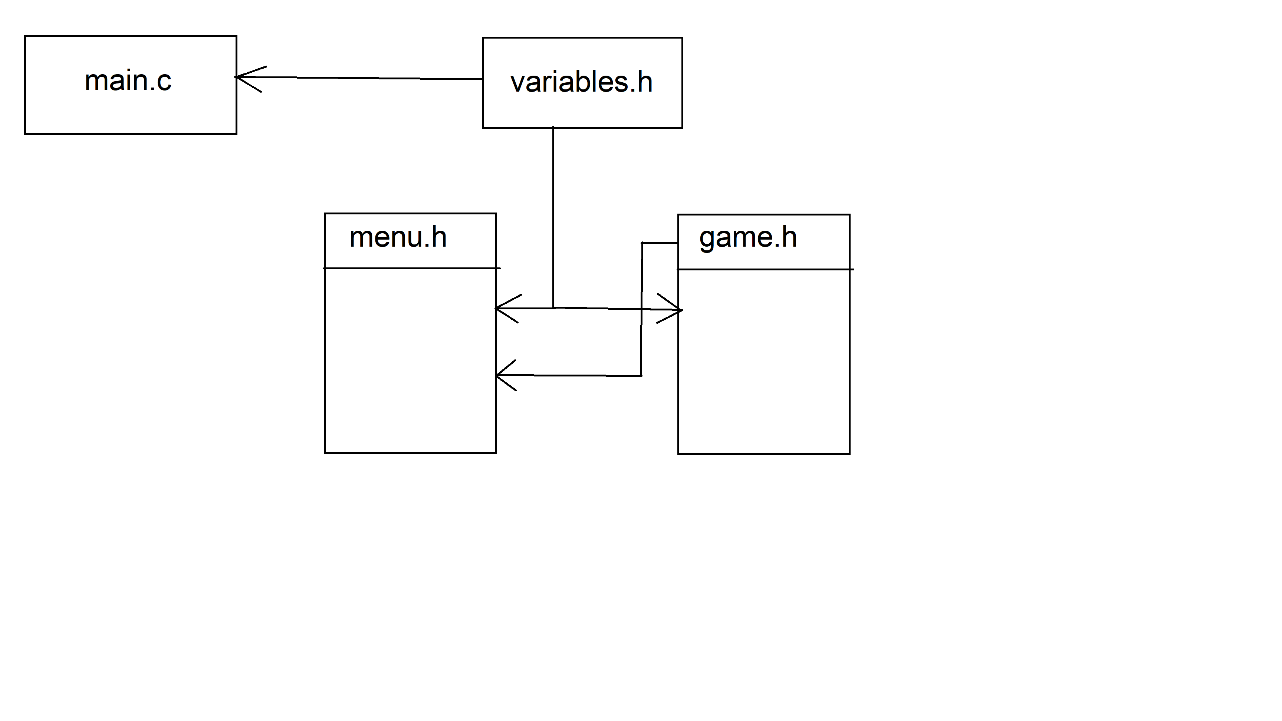
Також потрібно передбачити та забезпечити виконання всіх можливих розгалужень алгоритма, тобто програма повинна коректно реагувати на будь-які можливі ситуації (наприклад, виникнення помилкових ситуацій, перевірка файлів на порожність, правильність введених з клавіатури значень і т. д.). Передбачити взаємодію з користувачем (наприклад, можливість виводу правил гри, допомоги), таймер, лічильник числа ходів відповідно до поставленої в конкретному варіанті задачі.

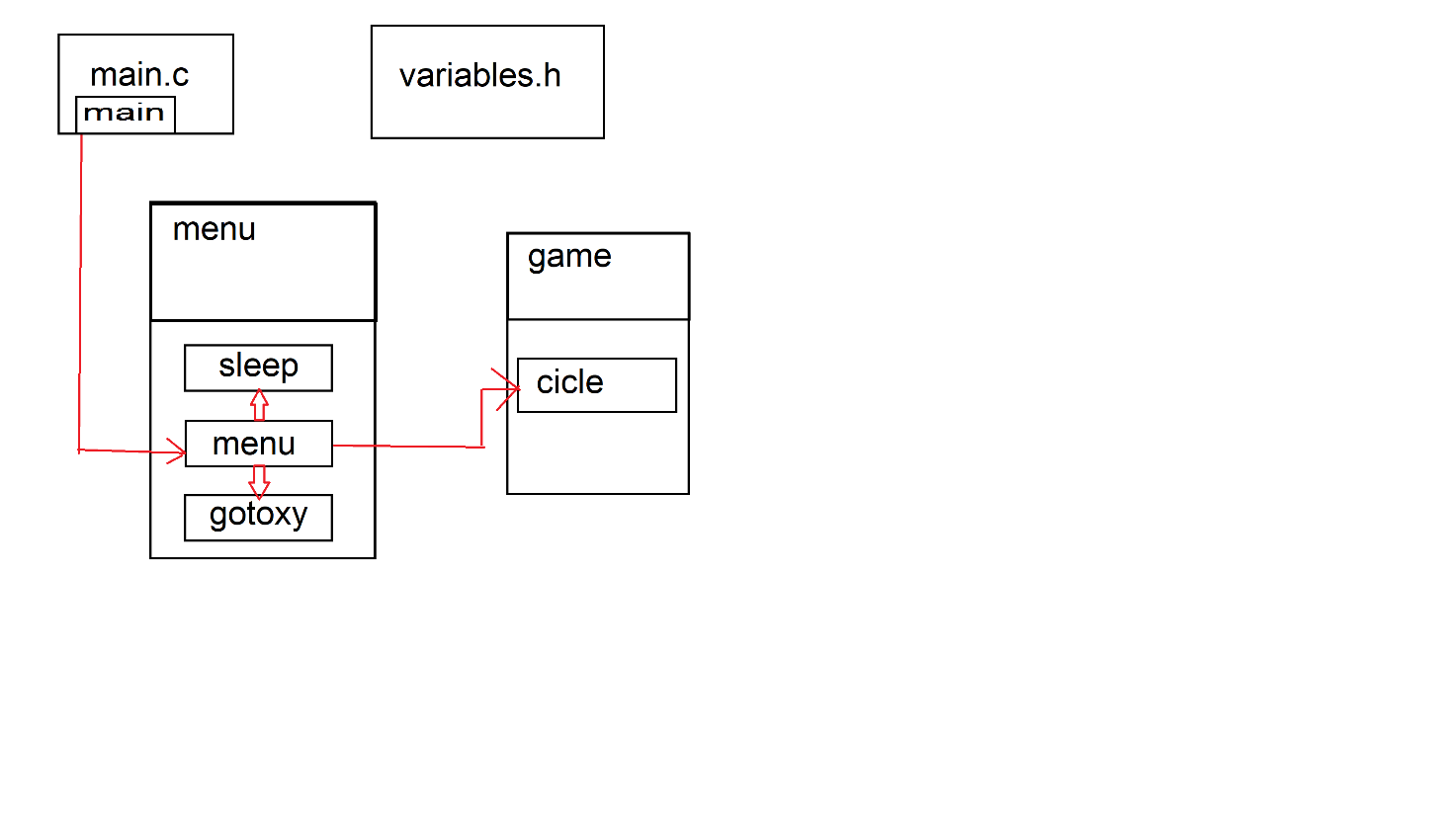
Варіант 12

«Гра в слова». Програма вибирає слово і малює на екрані стільки прочерків, скільки літер в слові. Відгадати це слово. Якщо називається літера, що входить в це слово, то вона підставляється замість відповідного прочерка. В противному випадку гравець втрачає один бал. На початок гри у гравця 15 балів.

Описання структури програми, тобто ілюстрація взаємозв’язків підпрограм в програмі.

1. **Схема імпорту/експорту модулів програми**

 **2)Структурна схема взаємовикликів процедур та функцій програми**



1. Блок main: перевірка файлу зі словами на порожність, запам’ятовування місць пробілів в масив, виклик меню
2. Блок menu: вивід на екран меню варіантами почати гру для одного слова, ряду слів і вийти.
3. Блок game: функція з циклом, який слідкує за кількістю балів, що залишилася у гравця, підциклом, що перевіяє чи введена буква відповідає букві, яка є у поточному слові
4. Блок variables: змінні для модулів

Текст програми з коментарями.

Вміст main.c

#include <stdio.h>

#include <stdlib.h>

#include "menu.h"

#include "variables.h"

#include <string.h>

**int** **main**()

{

probeli=**1**;

fil = fopen("words.txt", "rt");

**if** (fil == **0**)

{

perror("Can’t open file for reading"); //перевірка файл на порожність

}

**while**(fread(&ch,**sizeof**(ch),**1**,fil)) //знайдемо кількість пробелів для розміру масиву з місцями пробілів

{

**if**(ch==' ')

probeli++;

}

**int** probeli1[probeli]; //місця пробілів

rewind(fil); **int** mer=**0**, mer1=**0**;

**while**(fread(&ch,**sizeof**(ch),**1**,fil))

{

**if**(ch==' ')

{

probeli1[mer1]=mer; // запам'ятовуємо місця пробелів

mer1++;

}

mer++;

}

//здесь был цикл

menu(**75**/**2**, **10**, probeli1); //вивід меню

**return** **0**;

}

Вміст menu.c

#include "menu.h"

#include <stdio.h>

#include <windows.h>

#include <conio.h>

#include "variables.h"

**void** **gotoxy**(**const** **int** X,**const** **int** Y) //Встановлює курсор на (Х,Y), Х по горизонталі, Y по вертикальному положенню

{

HANDLE OutputHandle;

CONSOLE\_SCREEN\_BUFFER\_INFO ScreenBufInfo;

OutputHandle=GetStdHandle(STD\_OUTPUT\_HANDLE);

GetConsoleScreenBufferInfo(OutputHandle,&ScreenBufInfo);

ScreenBufInfo.dwCursorPosition.X=X;

ScreenBufInfo.dwCursorPosition.Y=Y;

SetConsoleCursorPosition(OutputHandle,ScreenBufInfo.dwCursorPosition);

}

**void** **sleep**(**int** ms) //створює паузу

{

**while** (ms -- > **0**)

usleep(**1000**);

}

**void** **menu**(**int** x, **int** y, **int** \*probeli1) //програма виводу головного меню

{

system("cls");//очищення екрана

**const** **int** n=**3**;

**int** i;

**typedef** **char** punkt[**50**];//введення типу змінної

punkt k[]= {"1) guess one random word", "2) guess words in a row", "3) exit"};

gotoxy((x-**5**), (y-**6**));

puts("GUESSING GAME");

**for** (i=**0**; i<=n-**1**; i++) //цикл для виведення пунктів меню

{

gotoxy(x-**7**, y-**2**+**2**\*i);

puts(k[i]);

}

sleep(**50**);

printf("**\n\n\n**If you want to continiue, make your choise(1, 2, or 3) : ");//діалогз користувачем

scanf("%i",&h);

system("cls");//очищення екрана

**if**(h!=**3**)

{

printf("Game is simple.**\n**1) You have 15 points for a game. **\n**");

printf("2) Every wrong letter is a lost point.**\n**");

printf("3) If you lose all the points - the game is over.**\n\n**");

**if** (h==**1**)

{

cicle(probeli1, **1**); //функція для одного випадкового слова

}

**else** **if** (h==**2**)

{

cicle(probeli1, **2**); //функція для ряду слів

}

sleep(**70**);

menu(**75**/**2**, **10**, probeli1);

}

**if** (h==**3**)

{

printf("End of the program.**\n**");

}

exit(**1**);//завершення роботи програми

}

Вміст menu.h

#ifndef \_MENU\_

#define \_MENU\_

**void** **gotoxy**(**const** **int** X,**const** **int** Y);//Встановлює курсор на (Х,Y), Х по горизонталі, Y по вертикальному положенню

**void** **sleep**(**int** ms);//створює паузу

**void** **menu**(**int** x, **int** y, **int** \*probeli1);//Виконує вивід головного меню

#endif

Вміст game.c

#include "game.h"

#include <stdio.h>

#include <stdlib.h>

#include "variables.h"

**void** **cicle**(**int** \*probeli1, **int** a)

{

**int** stock,s=**0**;

**int** other, some=**0**;

**if**(a==**1**) //варіант програми з одним словом

{

other=rand()%(probeli-**1**);

stock=other+**1**;

}

**else** **if**(a==**2**) //варіант програми з рядом слів

{

stock=probeli-**1**;

other=**0**;

}

**for**(**int** i=other; i<stock;i++)

{

**if**(s>=**15**) //перевірка чи кількість балів впала до 0 з 15

{

**break**;

}

**int** pro=probeli1[i]+**1**;

**int** letters=**0**;

rewind(fil);

fseek(fil, **sizeof**(**char**)\*pro, SEEK\_SET); //встановлюємо вказівник на поточне слово

**while**(fread(&ch,**sizeof**(ch),**1**,fil) && ch!=' ')//знайдемо кількість літер у слові для розміру масиву зі словом

{

letters++;

}

**if**(letters<**2**)

{

**break**;

}

**if**(some>**0**)

{

system("cls");

}

printf("**\n\n**%d) word:**\n**", i+**1**);

//printf("letters is %d", letters);

rewind(fil);

fseek(fil, **sizeof**(**char**)\*pro, SEEK\_SET);

**char** mas[letters]; **int** var=**0**;

mas[var]=ch;

**while**(fread(&ch,**sizeof**(ch),**1**,fil) && ch!=' ')

{

var++;

mas[var]=ch;

}

**char** m;

**char** mas1[strlen(mas)];

**int** cic;

**if**(letters<**5**)

{

cic=strlen(mas)-**1**;

}

**else**

{

cic=strlen(mas);

}

**for**(**int** i=**0**, j=**0**;i<cic;i++) // заповнення масива з невідомим словом, де замість літер штрихи

{

**if**(mas[i]!="**\0**")

{

**if**(mas[i]!=' ')

{

mas1[j]='|'; //заповнення додаткового масива прочерками

//printf("%c", mas1[j]);

j++;

}

//printf("%c", mas[i]);

}

**else**

{

**break**;

}

}

printf("**\n**");

**int** dop=strlen(mas1);

**for**(**int** i=**0**;i<strlen(mas1);i++)

{

printf("%c", mas1[i]);

}

**for**(**int** i=**0**, flag=**0**;i<dop;) //цикл для перевірки введеної летери на її місце в масиві

{

printf("**\n**");gets(&m);

**if**(s>=**15**)

{

**break**;

}

**int** f1=**0**;

printf("**\n**");

**for**(**int** j=**1**;j<=dop;j++)

{

**if**(m==mas[j] && mas1[j-**1**]!=m)

{

mas1[j-**1**]=m;

i++;

flag=**1**;

}

}

**if**(flag==**1**)

{

**for**(**int** i1=**0**;i1<dop;i1++)

{

printf("%c", mas1[i1]);

}

flag=**0**;

}

**else**

{

s++;

}

}

some++;

}

**return** **0**;

}

Вміст game.h

#ifndef \_GAME\_

#define \_GAME\_

**void** **cicle**(**int** \*probeli1, **int** a);

#endif

Вміст variables.h

#ifndef \_VAR\_

#define \_VAR\_

#include <stdio.h>

#include <stdlib.h>

**FILE** \*fil; **int** probeli; **char** ch;

**int** mer, mer1, h;

#endif

Вміст words.txt

summer day morning cat crown night midday