

# Основы программирования

## Лабораторная работа №19

Игра №2. Добавляем таблицу рекордов.

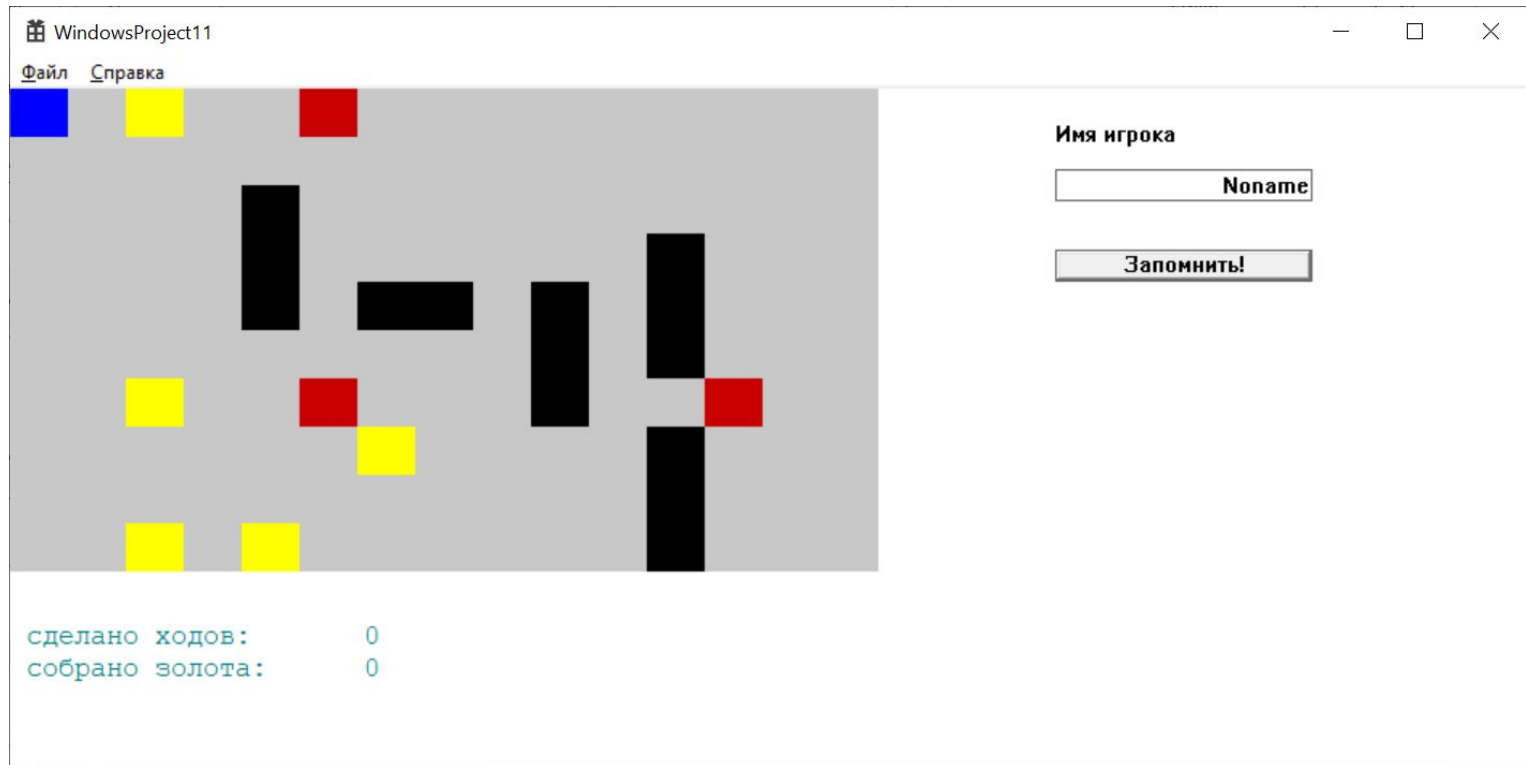
Массив записей.

Сортировка: вставкой, выбором, пузырьком.

Власенко Олег Федосович

# Таблица рекордов

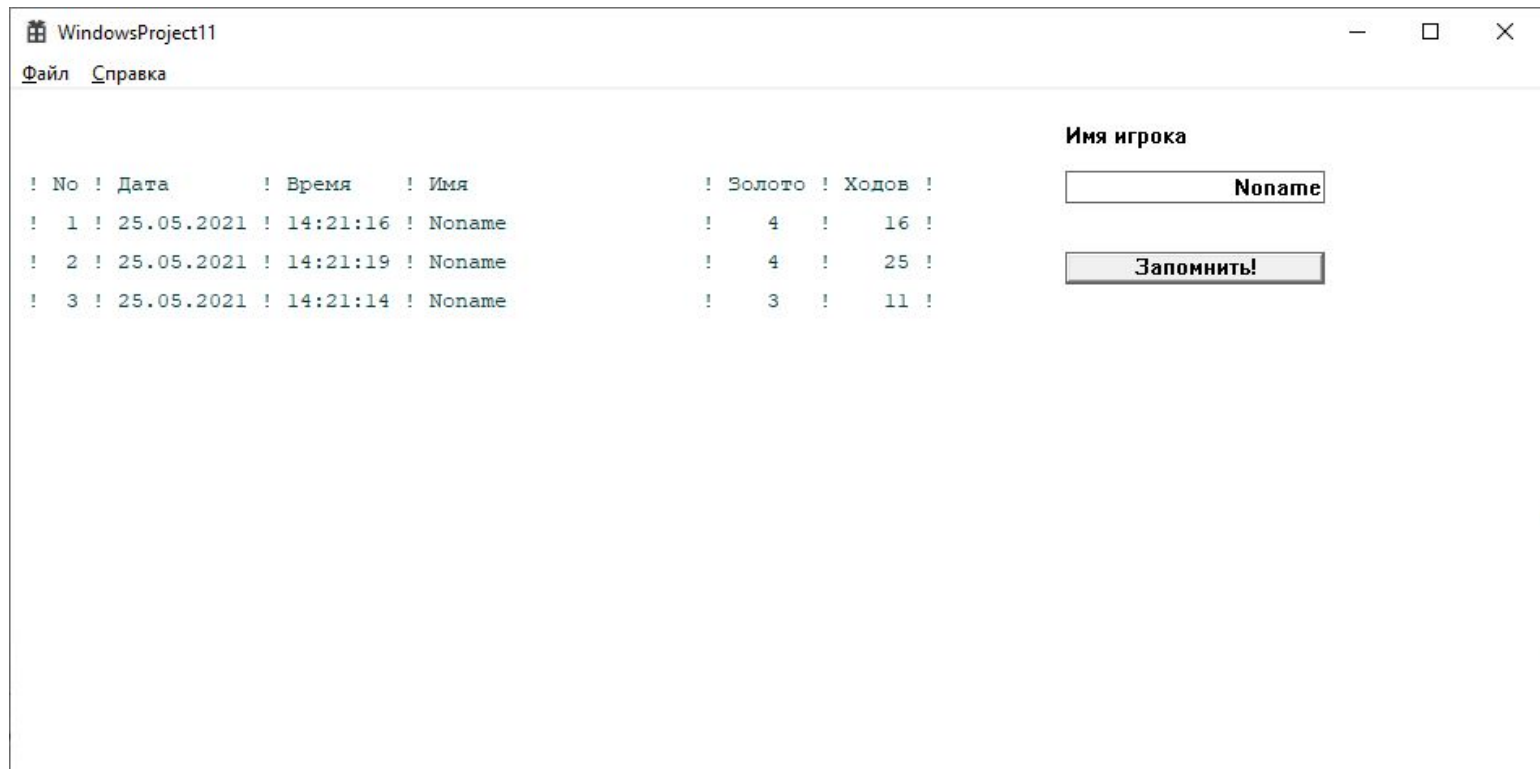
Добавим в игру таблицу рекордов.  
Поле для ввода имени и кнопка



# Таблица рекордов

Добавим в игру таблицу рекордов.

**Отображение лучших результатов в виде таблицы**



# Поле для ввода имени и кнопка - создание

```
LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)
{
    static HWND hBtn; // дескриптор кнопки
    static HWND hEdt1; // дескрипторы поля редактирования

    switch (message)
    {
        case WM_CREATE: // сообщение создания окна
            hInst = ((LPCREATESTRUCT)lParam)->hInstance; // дескриптор приложения
            // Создаем и показываем поле редактирования - для ввода имени рекордсмена
            hEdt1 = CreateWindowW(_T("edit"), _T("Noname"),
                WS_CHILD | WS_VISIBLE | WS_BORDER | ES_RIGHT, 650, 50, 160, 20,
                hWnd, 0, hInst, NULL);
            ShowWindow(hEdt1, SW_SHOWNORMAL);

            // Создаем и показываем кнопку
            hBtn = CreateWindowW(_T("button"), _T("Запомнить!"),
                WS_CHILD | WS_VISIBLE | WS_BORDER,
                650, 100, 160, 20, hWnd, 0, hInst, NULL);
            ShowWindow(hBtn, SW_SHOWNORMAL);

            SetTimer(hWnd, 1, 500, 0); // стартуем таймер - для движения монстров
            srand(time(NULL)); // Перезапуск генератора случайных чисел
            break;
    }
}
```

# Реакция на нажатие кнопки

```
case WM_COMMAND:
{
    if (lParam == (LPARAM)hBtn)    // если нажали на кнопку
    {
        TCHAR StrT[20];
        char str[20];

        // Берем имя из элемента редактирования и помещаем в строку Windows
        GetWindowText(hEdt1, StrT, sizeof(StrT));

        // Конвертирует строку Windows в строку Си
        // !!!! ВАЖНО - корректно работает ТОЛЬКО для латинских букв!
        wcstombs(str, StrT, 20);

        // Фокус возвращаем в игру
        // нажатия клавиш снова управляют игрой!
        SetFocus(hWnd);

        // добавляем рекорд в таблицу рекордов
        //addRecord(str); // новый рекорд просто вставляем снизу в таблицу
        InsertRecord(str); // новый рекорд вставляем в таблицу, сохраняя сортировку
        InvalidateRect(hWnd, NULL, TRUE);
    }
}
```

# Реакция на нажатие кнопки (2)

```
else {
    // Этот код был ранее - он обрабатывает выбор пунктов меню
    int wmId = LOWORD(wParam);
    // Разобрать выбор в меню:
    switch (wmId)
    {
        case IDM_ABOUT:
            DialogBox(hInst, MAKEINTRESOURCE(IDD_ABOUTBOX), hWnd, About);
            break;
        case IDM_EXIT:
            DestroyWindow(hWnd);
            break;
        default:
            return DefWindowProc(hWnd, message, wParam, lParam);
    }
}
} // case WM_COMMAND:
break;
```

# Управление клавиатурой

Переключение режимов отображения

```
// Новая глобальная переменная
int showMode = 1; // 1 - отображается игра,
                  // 0 - отображается таблица рекордов

...

// LRESULT CALLBACK WndProc(

case WM_KEYDOWN:
    switch (wParam)
    {

        case VK_ESCAPE:
            showMode = !showMode; // был 1 - будет 0, был 0 - будет 1
            InvalidateRect(hWnd, NULL, TRUE);
            break;
```

# Управление клавиатурой

Изменение порядка рекордов в таблице рекордов

```
case VK_F1:  
    SelectSortingDown();  
    InvalidateRect(hWnd, NULL, TRUE);  
    break;  
  
case VK_F2:  
    BubbleSortingUp();  
    //SelectSortingUp();  
    InvalidateRect(hWnd, NULL, TRUE);  
    break;
```



# Собственно таблица рекордов

```
// Структура с информацией о рекорде
struct Record {
    char name[20];
    int gold;
    int steps;
    unsigned int year;
    unsigned int month;
    unsigned int day;
    unsigned int hour;
    unsigned int minute;
    unsigned int second;
};

// Максимальное количество рекордов в таблице
#define MAX_NUM_RECORDS 10

// Таблица рекордов
struct Record records[MAX_NUM_RECORDS + 1];
// текущее количество рекордов в таблице
int numRecords = 0;
```

# Добавление рекорда в конец (без сортировки)

```
void addRecord(char name[])
{
    //if (numRecords >= MAX_NUM_RECORDS) {
    //numRecords = numRecords - 1;
    //}

    strcpy(records[numRecords].name, name);
    records[numRecords].gold = gold;
    records[numRecords].steps = steps;

    SYSTEMTIME st;
    // Получаем текущее время
    GetLocalTime(&st);

    // и разбрасываем его по полям в таблицу рекордов
    records[numRecords].year = st.wYear;
    records[numRecords].month = st.wMonth;
    records[numRecords].day = st.wDay;

    records[numRecords].hour = st.wHour;
    records[numRecords].minute = st.wMinute;
    records[numRecords].second = st.wSecond;
    // Следующий раз будем записывать рекорд в следующий элемент
    numRecords++;
}
```

# Сравнение двух рекордов

```
int CompareRecords(int index1, int index2)
{
    if (records[index1].gold < records[index2].gold)
        return -1;
    if (records[index1].gold > records[index2].gold)
        return +1;

    // if (records[index1].gold == records[index2].gold) {

    if (records[index1].steps > records[index2].steps)
        return -1;
    if (records[index1].steps < records[index2].steps)
        return +1;

    //   if (records[index1].steps == records[index2].steps) {
    return 0;
    //   }
    // }

}
```

# Добавление рекорда с сортировкой

```
void InsertRecord(char name[])
{
    strcpy(records[numRecords].name, name);
    records[numRecords].gold = gold;
    records[numRecords].steps = steps;

    SYSTEMTIME st;
    // Получаем текущее время
    GetLocalTime(&st);

    // и разбрасываем его по полям в таблицу рекордов
    records[numRecords].year = st.wYear;
    records[numRecords].month = st.wMonth;
    records[numRecords].day = st.wDay;

    records[numRecords].hour = st.wHour;
    records[numRecords].minute = st.wMinute;
    records[numRecords].second = st.wSecond;
```

# Добавление рекорда с сортировкой (2)

```
// Продвигаем запись к началу массива - если в ней
// хороший результат
int i = numRecords;
while (i > 0) {
    if (CompareRecords(i - 1, i) < 0) {
        struct Record temp = records[i];
        records[i] = records[i - 1];
        records[i - 1] = temp;
    }
    i--;
}
// Если таблица заполнена не полностью
if (numRecords < MAX_NUM_RECORDS)
    // следующий раз новый рекорд будет занесен в новый элемент
    numRecords++;
}
```

# Отображение таблицы рекордов (функция)

```
void DrawRecords(HDC hdc) {
    HFONT hFont;
    hFont = CreateFont(16, 0, 0, 0, 0, 0, 0, 0,
        DEFAULT_CHARSET, 0, 0, 0, 0,
        L"Courier New"
    );
    SelectObject(hdc, hFont);
    SetTextColor(hdc, RGB(0, 64, 64));

    TCHAR string1[] = _T("! No ! Дата      ! Время      ! Имя                                     ! Золото ! Ходов !");
    TextOut(hdc, 10, 50, (LPCWSTR)string1, _tcslen(string1));

    int i;
    for (i = 0; i < numRecords; i++) {
        TCHAR string2[80];
        char str[80];
        sprintf(str, "! %2d ! %02d.%02d.%04d ! %02d:%02d:%02d ! %-20s ! %4d      ! %5d !",
            i + 1,
            records[i].day, records[i].month, records[i].year,
            records[i].hour, records[i].minute, records[i].second,
            records[i].name, records[i].gold, records[i].steps
        );
        OemToChar(str, string2);
        TextOut(hdc, 10, 24 * (i + 1) + 50, (LPCWSTR)string2, _tcslen(string2));
    }
    DeleteObject(hFont);
}
```

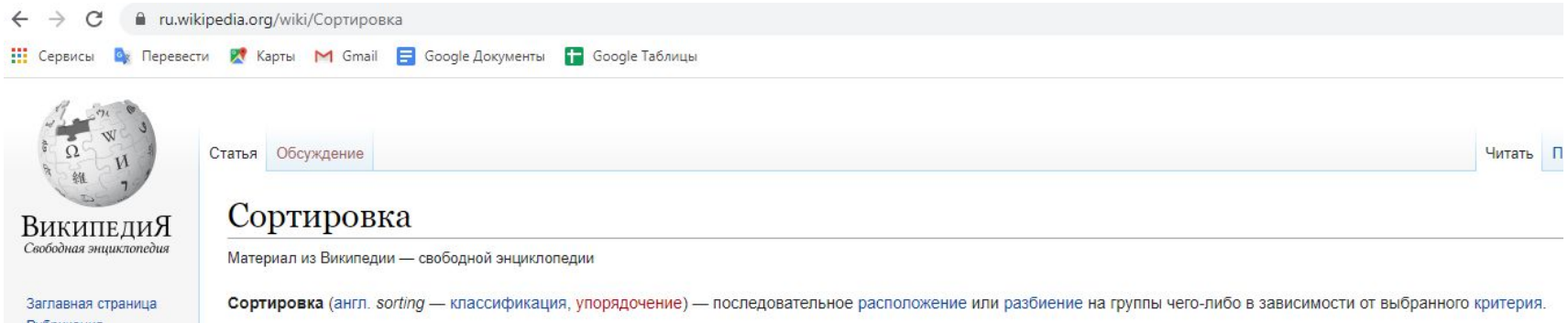
# Отображение таблицы рекордов (вызов)

```
case WM_PAINT:
{
    PAINTSTRUCT ps;
    HDC hdc = BeginPaint(hWnd, &ps);
    // TODO: Добавьте сюда любой код прорисовки, использующий
    HDC...

    TextOut(hdc, 650, 20, _T("Имя игрока"), 10);

    if (showMode == 1) {
        DrawField(hdc);
    }
    else
    {
        DrawRecords(hdc);
    }
    EndPaint(hWnd, &ps);
}
break;
```

# Сортировка (массива)



← → ↻ ru.wikipedia.org/wiki/Сортировка

Сервисы Перевести Карты Gmail Google Документы Google Таблицы

Статья **Обсуждение** Читать П

## Сортировка

Материал из Википедии — свободной энциклопедии

**Сортировка** (англ. *sorting* — классификация, упорядочение) — последовательное расположение или разбиение на группы чего-либо в зависимости от выбранного критерия.

Википедия  
Свободная энциклопедия

Заглавная страница  
Публикации

Базовые методы сортировки:

- Сортировка выбором
- Сортировка пузырьком
- Сортировка вставками



# Сортировка вставками

Google

сортировка вставками

Все Картинки Видео Новости Карты Ещё Настройки Инструменты

Результатов: примерно 706 000 (0,40 сек.)

[https://ru.wikipedia.org/wiki/Сортировка\\_вставками](https://ru.wikipedia.org/wiki/Сортировка_вставками)

**Сортировка вставками** — Википедия

Сортировка вставками (англ. Insertion sort) — алгоритм сортировки, в котором элементы входной последовательности просматриваются по одному, ...

Описание · Псевдокод · Анализ алгоритма

<https://habr.com/post/>

**В мире алгоритмов: Сортировка Вставками / Хабр**

28 мая 2013 г. — Сортировка вставками (Insertion Sort) — это простой алгоритм сортировки. Суть его заключается в том что, на каждом шаге алгоритма ...

Похожие запросы

- Как работает сортировка вставками?
- Как работает сортировка бинарными вставками?
- Что такое бинарная вставка?
- Чем характеризуется алгоритм сортировки вставкой?

[Оставить отзыв](#)

<https://habr.com/post/>

**Сортировки вставками / Хабр**

2 июл. 2018 г. — Сортировки вставками. Python.; Программирование.; Java.; Алгоритмы.; Визуализация данных. Общая суть ...

- [https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0\\_%D0%B2%D1%81%D1%82%D0%B0%D0%B2%D0%BA%D0%B0%D0%BC%D0%B8](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0_%D0%B2%D1%81%D1%82%D0%B0%D0%B2%D0%BA%D0%B0%D0%BC%D0%B8)
- <https://habr.com/ru/post/181271/>
- Реализации алгоритмов/Сортировка/Вставками - [https://ru.wikibooks.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8\\_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%BE%D0%B2/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0/%D0%92%D1%81%D1%82%D0%B0%D0%B2%D0%BA%D0%B0%D0%BC%D0%B8](https://ru.wikibooks.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%BE%D0%B2/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0/%D0%92%D1%81%D1%82%D0%B0%D0%B2%D0%BA%D0%B0%D0%BC%D0%B8)

# Сортировка пузырьком



сортировка пузырьком

Все Картинки Видео Новости Покупки Ещё Настройки Инструменты

Результатов: примерно 25 300 (0,40 сек.)

[https://ru.wikipedia.org/wiki/Сортировка\\_пузырьком](https://ru.wikipedia.org/wiki/Сортировка_пузырьком)

**Сортировка пузырьком** — Википедия

Сортировка простыми обменами, **сортировка пузырьком** (англ. bubble sort) — простой алгоритм сортировки. Для понимания и реализации этот ...  
Алгоритм · Реализация · Пример работы алгоритма

[http://algotlist.ru/sort/bubble\\_sort](http://algotlist.ru/sort/bubble_sort)

**Сортировка пузырьком** - AlgoList

Идея метода: шаг **сортировки** состоит в проходе снизу вверх по массиву. По пути просматриваются пары соседних элементов. Если элементы некоторой ...

[https://neerc.ifmo.ru/wiki?title=Сортировка\\_пузырьком](https://neerc.ifmo.ru/wiki?title=Сортировка_пузырьком)

**Сортировка пузырьком** — Викиконспекты

Сортировка простыми обменами, **сортировка пузырьком** (англ. bubble sort) — один из квадратичных алгоритмов сортировки. Алгоритм[править]. Алгоритм ...

<https://habr.com/post/>

**Пузырьковая сортировка и все-все-все** / Хабр

3 дек. 2013 г. — В обычном «**пузырьке**» во время каждого прохода мы планомерно выдавливаем в конец массива текущий максимум. Если же ...

- [https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0\\_%D0%BF%D1%83%D0%B7%D1%8B%D1%80%D1%8C%D0%BA%D0%BE%D0%BC](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0_%D0%BF%D1%83%D0%B7%D1%8B%D1%80%D1%8C%D0%BA%D0%BE%D0%BC)
- [http://algotlist.ru/sort/bubble\\_sort.php](http://algotlist.ru/sort/bubble_sort.php)
- Реализации алгоритмов/Сортировка/Пузырьком-  
[https://ru.wikibooks.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8\\_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%BE%D0%B2/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0/%D0%9F%D1%83%D0%B7%D1%8B%D1%80%D1%8C%D0%BA%D0%BE%D0%BC](https://ru.wikibooks.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%BE%D0%B2/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0/%D0%9F%D1%83%D0%B7%D1%8B%D1%80%D1%8C%D0%BA%D0%BE%D0%BC)

# Сортировка выбором



сортировка выбором



Все   Картинки   Видео   Новости   Карты   Ещё   Настройки   Инструменты

Результатов: примерно 1 040 000 (0,73 сек.)

[https://ru.wikipedia.org/wiki/Сортировка\\_выбором](https://ru.wikipedia.org/wiki/Сортировка_выбором)

Сортировка выбором — Википедия

**Сортировка выбором** (Selection sort) — алгоритм сортировки. Может быть как устойчивый, так и неустойчивый. На массиве из  $n$  элементов имеет время ...

<https://habr.com/post/>

Сортировки выбором / Хабр - Habr

3 сент. 2018 г. — Ну, или сортировки вставками — разновидность сортировок выбором. Или сортировки выбором — частный случай сортировок вставками ...

Похожие запросы

Как работает сортировка выбором?

Какая сложность у алгоритма сортировки выбором?

Каким образом выполняется сортировка простым выбором?

В чем суть сортировки методом выбора?

- [https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0\\_%D0%B2%D1%8B%D0%B1%D0%BE%D1%80%D0%BE%D0%BC](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0_%D0%B2%D1%8B%D0%B1%D0%BE%D1%80%D0%BE%D0%BC)
- <https://habr.com/ru/post/422085/>
- Реализации алгоритмов/Сортировка/Выбором -  
[https://ru.wikibooks.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8\\_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%BE%D0%B2/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0/%D0%92%D1%8B%D0%B1%D0%BE%D1%80%D0%BE%D0%BC](https://ru.wikibooks.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%BE%D0%B2/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0/%D0%92%D1%8B%D0%B1%D0%BE%D1%80%D0%BE%D0%BC)

# Реализация сортировки выбором для рекордов

```
void SelectSortingDown() {
    int i;
    for (i = 0; i < numRecords - 1; i++) {
        int indexMax = i;
        int j;
        for (j = i + 1; j < numRecords; j++) {
            if (CompareRecords(j, indexMax) > 0) {
                indexMax = j;
            }
        }

        if (indexMax != i) {
            struct Record temp = records[i];
            records[i] = records[indexMax];
            records[indexMax] = temp;
        }
    }
}
```

# Реализация сортировки пузырьком для рекордов

```
void BubbleSortingUp() {
    int i;
    for (i = 0; i < numRecords - 1; i++) {
        int j;
        for (j = 0; j < numRecords - 1; j++) {
            if (CompareRecords(j, j + 1) > 0) {
                struct Record temp = records[j];
                records[j] = records[j + 1];
                records[j + 1] = temp;
            }
        }
    }
}
```

# Задача 1

Реализовать сортировку по возрастанию, используя один из алгоритмов:

- 1) Сортировка вставкой
- 2) Сортировка пузырьком
- 3) Сортировка выбором

# Задача 2

Реализовать сортировку по убыванию, используя один из алгоритмов:

- 1) Сортировка вставкой
- 2) Сортировка пузырьком
- 3) Сортировка выбором

# Задача 3

Реализовать программу обеспечивающую выполнение следующих действий:

- 1) Ввод массива с клавиатуры (или инициализация в самой программе)
- 2) Вывод текущего состояния массива
- 3) Сортировка по возрастанию (Задача 1)
- 4) Вывод текущего состояния массива
- 5) Сортировка по убыванию (Задача 2)
- 6) Вывод текущего состояния массива



# Полезные ссылки

- **Элементы управления окна** - <https://prog-cpp.ru/winelements/>
- **Как преобразовать указатель tchar в указатель char** - <https://coderoad.ru/1721731/%D0%9A%D0%B0%D0%BA-%D0%BF%D1%80%D0%B5%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D1%8C-%D1%83%D0%BA%D0%B0%D0%B7%D0%B0%D1%82%D0%B5%D0%BB%D1%8C-tchar-%D0%B2-%D1%83%D0%BA%D0%B0%D0%B7%D0%B0%D1%82%D0%B5%D0%BB%D1%8C-char> **(Работает для латиницы!)**
- **Получение даты и времени с помощью Win32 API** - <https://gamedev.ru/code/forum/?id=130983>
- **SetFocus function** - <https://docs.microsoft.com/en-us/windows/win32/api/winuser/nf-winuser-setfocus>