

**Федеральное государственное автономное образовательное
учреждение высшего образования "Национальный исследовательский
университет "Высшая школа экономики""**

**Московский институт электроники и математики им. А.Н. Тихонова
НИУ ВШЭ**

Департамент компьютерной инженерии

Курс: Алгоритмизация и программирование

Отчёт

по лабораторной работе №5

Раздел	Макс оценка	Итог. оценка
Работа программы	1	
Тесты	1	
Правильность алгоритма	3	
Ответы на вопросы	2	
Доп.задание	3	
Итого		

Студент: Андреев Глеб Андреевич

Группа: БИВ254

Вариант: №252 (1, 6, 11)

**Руководитель: Альбатша Ахмад
Мухаммад Хусайн**

Оценка: _____

Дата сдачи: _____

Оглавление

<i>Задание</i>	3
<i>Листинг</i>	4
<i>Задание 1.</i>	4
<i>Задание 2</i>	6
<i>Тесты</i>	9
<i>Задание 1.</i>	9
<i>Задание 2.</i>	9

Задание

ЦЕЛЬ РАБОТЫ: Программирование с использованием структурированных типов данных (файлы). Динамические структуры данных.

- I. (1) Создать файл для хранения действительных чисел, вводимых с клавиатуры. Прочитать этот файл и вычислить минимальное положительное число;
- II. Создать связанный список для хранения целых чисел. Число записей неизвестно, но можно ввести число, являющееся признаком окончания ввода данных. Для четных номеров вариантов организовать очередь, а для нечетных – стек.
 - (6) Для исходного списка решить следующую задачу: вставить заданное число A_1 после каждого положительного элемента
 - (11) Для полученного списка решить следующую задачу: упорядочить по возрастанию методом "пузырька" элементы списка, расположенные до первого четного элемента.

ПРИМЕЧАНИЕ. Каждую часть оформить как отдельную программу.

ЗАМЕЧАНИЕ Все части оформить в одной программе, для каждой части написать отдельную функцию. Предусмотреть вывод списка после выполнения каждого задания с помощью отдельной функции .

Листинг

Задание 1.

```
#include <stdio.h>
#include <stdlib.h>
#include <limits.h>

#define fname "numbers.txt"

int minPositive(double *res) {
    FILE *f = fopen(fname, "r");
    if (!f) return 0;

    double x, min = INT_MAX;
    int found = 0;

    while (fscanf(f, "%lf", &x) == 1) {
        if (x > 0 && x < min) {
            min = x;
            found = 1;
        }
    }

    fclose(f);

    if (found) *res = min;
    return found;
}

int file_write() {
    int len = 0;
    double x;
    char check;
    FILE *f = fopen(fname, "w");
    if (!f) {
        printf("Cannot create file.\n");
        return 0;
    }

    printf("Enter real numbers for file (q to stop):\n");

    while (1) {
        if (scanf("%lf", &x) == 1) {
            fprintf(f, "%lf ", x);
            len++;
        } else {
            check = getchar();
            if (check == 'q') break;
        }
    }
}
```

```
fclose(f);
return len;
}

int main() {
    printf("Lab work. Part I. Task 1\n");

    double result;
    int len = file_write();

    if (len == 0) {
        printf("No numbers were written to the file.\n");
        return 0;
    }

    if (!minPositive(&result)) {
        printf("\nNo positive numbers in file.\n");
        return 0;
    }

    printf("\nMinimal positive number: %lf\n", result);

    return 0;
}
```

Задание 2

```
#include <stdio.h>
#include <stdlib.h>
#include <limits.h>

typedef struct node {
    int x;
    struct node *next;
} node;

node* pushStack(node *head, int x) {
    node *p = (node*)malloc(sizeof(node));
    p->x = x;
    p->next = head;
    return p;
}

node* pushQueue(node *head, int x) {
    node *p = (node*)malloc(sizeof(node));
    p->x = x;
    p->next = NULL;

    if (!head) return p;

    node *q = head;
    while (q->next) q = q->next;
    q->next = p;

    return head;
}

void printList(node *head) {
    node *p = head;
    while (p) {
        printf("%d ", p->x);
        p = p->next;
    }
    printf("\n");
}

int insertAfterPositive(node *head, int A1) {
    node *p = head;
    int flag = 0;
    while (p) {
        if (p->x > 0) {
            node *q = (node*)malloc(sizeof(node));
            q->x = A1;
            q->next = p->next;
            p->next = q;
            p = q->next;
            flag = 1;
        }
    }
}
```

```

        else {
            p = p->next;
        }
    }
    return flag;
}

void bubbleSortBeforeEven(node *head) {
    if (!head) return;

    node *stop = head;

    while (stop && stop->x % 2 != 0)
        stop = stop->next;

    if (stop == head) return;

    int swapped = 1;
    while (swapped) {
        swapped = 0;
        node *p = head;

        while (p->next && p->next != stop) {
            if (p->x > p->next->x) {
                int t = p->x;
                p->x = p->next->x;
                p->next->x = t;
                swapped = 1;
            }
            p = p->next;
        }
    }
}

int main() {
    printf("Lab Work #5. Task 2\n");
    node *head = NULL;
    int x;
    char check;

    printf("\nInput integers (-999 ends input):\n");

    while (1) {
        do {
            scanf("%d", &x);
            check = getchar();
            if (check != '\n') {
                printf("Invalid input.\n");
                x = INT_MAX;
                while (check != '\n' && check != EOF) {
                    check = getchar();
                }
            }
        }
    }
}

```

```

        }
    } while (check != '\n' || x == INT_MAX);

    if (x == -999) break;
    head = pushStack(head, x);
}

int A1;
printf("\nEnter A1: ");
do {
    scanf("%d", &A1);
    check = getchar();
    if (check != '\n') {
        printf("Invalid input.\n");
        A1 = INT_MAX;
        while (check != '\n' && check != EOF) {
            check = getchar();
        }
    }
} while (check != '\n' || A1 == INT_MAX);

printf("\nInitial list:\n");
printList(head);

if (insertAfterPositive(head, A1)) {
    printf("\nNew list:\n");
    printList(head);
}
else
    printf("No positive elements found in the list.\n");

bubbleSortBeforeEven(head);

printf("\nSorted list:\n");
printList(head);

return 0;
}

```

Тесты

Задание 1.

№	Исходные данные	Результаты
1	“numbers.txt” = “”	No numbers were written to the file.
2	“numbers.txt” = “1 3 -123 0.0 123.1 0.004”	Minimal positive number: 0.004
3	“numbers.txt” = “-1 -2 -3 -4”	No positive numbers in file.

Задание 2.

№	Исходные данные	Результаты
1	List = [-3 1 5 7 -11 2] A1 = 1	Initial list: -3 1 5 7 -11 2 New list: -3 1 1 5 1 7 1 -11 2 1 Sorted list: -11 -3 1 1 1 1 5 7 2 1
2	List = [-1 -3 -5 -100] A1 = 3	Initial list: -1 -3 -5 -100 No positive elements found in the list. Sorted list: -5 -3 -1 -100