

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»
(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных
систем

Лабораторная работа №12

по дисциплине: Основы программирования
тема: «Бинарный поиск»

Выполнил: студент группы ПВ-223
Мелехов Артём Дмитриевич

Проверили:
ст. преп. Притчин Иван Сергеевич
асс. Черников Сергей Викторович

Белгород 2023 г.

Лабораторная работа №12 «Бинарный поиск»

Цель работы: получение навыков использования алгоритмов бинарного поиска для решения задач оптимизации.

Содержание отчета:

Тема лабораторной работы

Цель лабораторной работы

Решения задач. Для каждой задачи указаны:

- Название задачи.
- Для задач со звездочкой приложены блок-схемы.
- Исходный код.
- Скриншот с *codeforces* с указанием вердикта тестирующей системы.

Вывод.

Задача №1. Двоичный поиск.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>

void inputArray(int* a, const size_t n)
{
    for (int i = 0; i < n; i++)
        scanf("%d", &a[i]);
}

int binarySearchLastEqual(const int* const a,
    const size_t n, const int x)
{
    int left = -1;
    int right = n;

    while (right - left > 1)
    {
        int middle = left + (right - left) / 2;

        if (a[middle] <= x)
            left = middle;
        else
            right = middle;
    }

    return (left != -1 && a[left] == x) ? left : -1;
}

int main(void)
{
    int n, k;
    scanf("%d %d", &n, &k);

    int* a = (int*)malloc(n * sizeof(int));
    inputArray(a, n);

    for (int i = 0; i < k; i++)
    {
        int request;
        scanf("%d", &request);

        printf(binarySearchLastEqual(a, n, request) ==
            -1 ? "NO\n" : "YES\n");
    }

    free(a);

    return 0;
}
```

Вердикт тестирующей системы:

177865512	25.10.2022 14:33	Saryi_Sky	А - Двоичный поиск	GNU C11	Полное решение	436 мс	400 КБ
---------------------------	---------------------	-----------	--	------------	-------------------	--------	--------

Задача №2. Ближайшее слева.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>

void inputArray(int* a, const size_t n)
{
    for (int i = 0; i < n; i++)
        scanf("%d", &a[i]);
}

int binarySearchTheMaxIndexIsNoMoreThenX
(const int* const a, const size_t n, const int x)
{
    int left = -1;
    int right = n;

    while (right - left > 1)
    {
        int middle = (right + left) / 2;

        if (a[middle] <= x)
            left = middle;
        else
            right = middle;
    }

    return ++left;
}

int main(void)
{
    int n, k;
    scanf("%d %d", &n, &k);

    int* a = (int*)malloc(n * sizeof(int));
    inputArray(a, n);

    for (int i = 0; i < k; i++)
    {
        int request;
        scanf("%d", &request);

        printf("%d\n", binarySearchTheMaxIndexIsNoMoreThenX
            (a, n, request));
    }

    free(a);

    return 0;
}
```

Вердикт тестирующей системы:

177869886	25.10.2022 15:09	Staryi_Sky	В - Ближайшее слева	GNU C11	Полное решение	468 мс	400 КБ
---------------------------	---------------------	----------------------------	---	------------	-------------------	--------	--------

Задача №3. Ближайшее справа.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>

void inputArray(int* a, const size_t n)
{
    for (int i = 0; i < n; i++)
        scanf("%d", &a[i]);
}

int binarySearchTheMinIndexIsAtLeastX
(const int* const a, const size_t n, const int x)
{
    int left = -1;
    int right = n;

    while (right - left > 1)
    {
        int middle = (right + left) / 2;

        if (a[middle] < x)
            left = middle;
        else
            right = middle;
    }

    return ++right;
}

int main(void)
{
    int n, k;
    scanf("%d %d", &n, &k);

    int* a = (int*)malloc(n * sizeof(int));
    inputArray(a, n);

    for (int i = 0; i < k; i++)
    {
        int request;
        scanf("%d", &request);

        printf("%d\n", binarySearchTheMinIndexIsAtLeastX
            (a, n, request));
    }

    free(a);

    return 0;
}
```

Вердикт тестирующей системы:

177870347	25.10.2022 15:13	Saryi_Sky	<u>C++</u> Ближайшее справа	GNU C11	Полное решение	436 мс	400 КБ
---------------------------	---------------------	-----------	-----------------------------------	------------	-------------------	--------	--------

Задача №4. Быстрый поиск в массиве.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>
#include <stdlib.h>

void inputArray(int* a, const size_t n)
{
    for (int i = 0; i < n; i++)
        scanf("%d", &a[i]);
}

int compare_ints(const void* a, const void* b)
{
    int arg1 = *(const int*)a;
    int arg2 = *(const int*)b;

    if (arg1 < arg2)
        return -1;

    if (arg1 > arg2)
        return 1;

    return 0;
}

int binarySearchTheMinIndexIsAtLeastX
(const int* const a, const size_t n, const int x)
{
    int left = -1;
    int right = n;

    while (right - left > 1)
    {
        int middle = (right + left) / 2;

        if (a[middle] < x)
            left = middle;
        else
            right = middle;
    }

    return right + 1;
}
```

```

int binarySearchTheMaxIndexIsNoMoreThenX
(const int* const a, const size_t n, const int x)
{
    int left = -1;
    int right = n;

    while (right - left > 1)
    {
        int middle = (right + left) / 2;

        if (a[middle] <= x)
            left = middle;
        else
            right = middle;
    }

    return left + 1;
}

int main(void)
{
    int n;
    scanf("%d", &n);

    int* a = (int*)malloc(n * sizeof(int));
    inputArray(a, n);

    qsort(a, n, sizeof(int), compare_ints);

    int k;
    scanf("%d", &k);

    for (int i = 0; i < k; i++)
    {
        int lRequest, rRequest;
        scanf("%d %d", &lRequest, &rRequest);

        int minLeft = binarySearchTheMinIndexIsAtLeastX
                        (a, n, lRequest);
        int maxRight = binarySearchTheMaxIndexIsNoMoreThenX
                        (a, n, rRequest);
        int result = maxRight - minLeft + 1;

        printf("%d ", result);
    }

    free(a);

    return 0;
}

```

Вердикт тестирующей системы:

177874246	25.10.2022 15:42	Staryi_Sky	D - Быстрый поиск в массиве	GNU C11	Полное решение	499 мс	400 КБ
---------------------------	---------------------	------------	---	------------	-------------------	--------	--------

Задача №5. Верёвочки.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>
#define EPS 1e-7

void inputArray(int* a, const size_t n)
{
    for (int i = 0; i < n; i++)
        scanf("%d", &a[i]);
}

long long countRopes(int* ropeLengths, size_t n,
                    double oneRopeLength)
{
    long long count = 0;

    for (size_t i = 0; i < n; i++)
        count += ropeLengths[i] / oneRopeLength;

    return count;
}

double maxLenOfPieces(const int* const ropeLengths,
                    const size_t n, const int k)
{
    double left = 0;
    double right = 1e7;

    while (right - left > EPS)
    {
        double middle = (right + left) / 2;

        if (countRopes(ropeLengths, n, middle) >= k)
            left = middle;
        else
            right = middle;
    }

    return left;
}

int main(void)
{
    int n, k;
    scanf("%d %d", &n, &k);

    int* ropeLengths = (int*)malloc(n * sizeof(int));
    inputArray(ropeLengths, n);

    printf("%f", maxLenOfPieces(ropeLengths, n, k));

    free(ropeLengths);

    return 0;
}
```

Вердикт тестирующей системы:

177910352	25.10.2022 20:28	Staryi_Sky	В - Веревочки	GNU C11	Полное решение	15 мс	0 КБ
---------------------------	---------------------	------------	-----------------------------------	------------	-------------------	-------	------

Задача №6. Очень Лёгкая Задача.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

void order2(long long* a, long long* b)
{
    if (*a > *b)
    {
        long long t = *a;
        *a = *b;
        *b = t;
    }
}

long long countCopies(long long firstXeroxTime,
                      long long secondXeroxTime, const long long sumTime)
{
    order2(&firstXeroxTime, &secondXeroxTime);

    return sumTime / firstXeroxTime +
           (sumTime - firstXeroxTime) / secondXeroxTime;
}

long long timeForNCopies(const long long n,
                        const long long x, const long long y)
{
    long long left = 0;
    long long right = 1e18;

    while (right - left > 1)
    {
        long long middle = (right + left) / 2;

        if (countCopies(x, y, middle) >= n)
            right = middle;
        else
            left = middle;
    }

    return right;
}

int main(void)
{
    long long n, x, y;
    scanf("%lld %lld %lld", &n, &x, &y);

    printf("%lld", timeForNCopies(n, x, y));

    return 0;
}
```

Вердикт тестирующей системы:

177912476	25.10.2022 20:48	Saryi_Sky	C - Очень Лёгкая Задача	GNU C11	Полное решение	15 мс	0 КБ
---------------------------	---------------------	-----------	---	------------	-------------------	-------	------

Задача №7. Ночная работа.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

long long sumCounter(long long n, long long middle,
                    long long k)
{
    long long sum = middle;
    long long multiplierK = k;

    while (sum <= n && k <= middle)
    {
        sum += (middle / k);
        k *= multiplierK;
    }

    return sum;
}

long long binarySearch(long long n, long long k)
{
    if (n < k)
        return n;

    long long left = k - 1;
    long long right = n + 1;

    while ((right - left) > 1)
    {
        long long middle = (left + right) / 2;

        if (sumCounter(n, middle, k) >= n)
            right = middle;
        else
            left = middle;
    }

    return right;
}

int main(void)
{
    long long n, k;
    scanf("%lld %lld", &n, &k);

    printf("%lld", binarySearch(n, k));

    return 0;
}
```

Вердикт тестирующей системы:

178110397	27.10.2022 17:02	Stanyi_Sky	165B - Ночная работа	GNU C11	Полное решение	60 мс	0 КБ
---------------------------	---------------------	------------	--------------------------------------	------------	----------------	-------	------

Задача №8. Компьютерная игра.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

int main(void)
{
    long long q;
    scanf("%lld", &q);

    for (int i = 0; i < q; i++)
    {
        long long k, n, a, b;
        scanf("%lld %lld %lld %lld", &k, &n, &a, &b);

        k -= n * a;

        if (k > 0)
            printf("%lld\n", n);
        else
        {
            k = -k + 1;

            long long difference = a - b;
            long long turns = (k + difference - 1) /
                               difference;

            if (turns > n)
                printf("-1\n");
            else
                printf("%lld\n", n - turns);
        }
    }

    return 0;
}
```

Вердикт тестирующей системы:

178278813	29.10.2022 00:09	Staryi_Sky	1183С - Компьютерная игра	GNU C11	Полное решение	499 мс	0 КБ
---------------------------	---------------------	------------	---	------------	----------------	--------	------

Задача №9. Книги.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>

void inputArray(long long* a, const size_t n)
{
    for (long long i = 0; i < n; i++)
        scanf("%lld", &a[i]);
}

long long max2(long long a, long long b)
{
    return a > b ? a : b;
}

long long binSearch(const long long* const b,
    const size_t n, const long long limiter)
{
    long long result = 0;

    for (long long start = 0; start < n; start++)
    {
        long long left = start;
        long long right = n + 1;

        while (right - left > 1)
        {
            long long middle = (left + right) / 2;

            if (b[middle] - b[start] <= limiter)
                left = middle;
            else
                right = middle;
        }

        result = max2(result, left - start);
    }

    return result;
}
```

```

int main(void)
{
    long long n, t;
    scanf("%lld %lld", &n, &t);

    long long* a = (long long*)malloc(sizeof(long long) *
                                         n);
    inputArray(a, n);

    long long* b = (long long*)malloc(sizeof(long long) *
                                         (n + 1));
    b[0] = 0;

    for (long long i = 1; i <= n; i++)
        b[i] = a[i - 1] + b[i - 1];

    printf("%lld", binSearch(b, n, t));

    free(a);
    free(b);

    return 0;
}

```

Вердикт тестирующей системы:

179414113	05.11.2022 12:50	Staryi_Sky	279В - Книги	GNU C11	Полное решение	92 мс	1600 КБ
---------------------------	---------------------	------------	------------------------------	------------	----------------	-------	---------

Задача №10. Евгений и плейлист.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>

void inputArray(long long* a, const size_t n)
{
    a[0] = 0;

    for (size_t i = 1; i <= n; i++)
    {
        long long c, t;
        scanf("%lld %lld", &c, &t);

        a[i] = c * t + a[i - 1];
    }
}

int main(void)
{
    long long n, m;
    scanf("%lld %lld", &n, &m);

    long long* a = (long long*)malloc(sizeof(long long) *
                                         n + 1);
    inputArray(a, n);

    for (long long i = 0; i < m; i++)
    {
        long long v;
        scanf("%lld", &v);

        long long left = -1;
        long long right = n + 1;

        while (right - left > 1)
        {
            long long middle = (left + right) / 2;

            if (a[middle] < v)
                left = middle;
            else
                right = middle;
        }

        printf("%lld\n", right);
    }

    free(a);

    return 0;
}
```

Вердикт тестирующей системы:

179451749	05.11.2022 18:35	Saryi_Sky	302B - Евгений и плейлист	GNU C11	Полное решение	966 мс	800 КБ
---------------------------	---------------------	-----------	---	------------	----------------	--------	--------

Задача №11. Алёна и узкий холодильник.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>
#include <stdlib.h>

void inputArray(long long* a, const size_t n)
{
    for (size_t i = 0; i < n; i++)
        scanf("%lld", &a[i]);
}

void arrayCopy(long long* copyA, const long long* a,
               size_t size)
{
    while (size--)
        *copyA++ = *a++;
}

int compareInts(const void* a, const void* b)
{
    int arg1 = *(const int*)a;
    int arg2 = *(const int*)b;

    if (arg1 < arg2)
        return -1;
    if (arg1 > arg2)
        return 1;

    return 0;
}

void swap(long long* a, long long* b)
{
    long long t = *a;
    *a = *b;
    *b = t;
}

void reverseArray(long long* a, const size_t n)
{
    for (size_t i = 0, j = n - 1; i < j; i++, j--)
        swap(&a[i], &a[j]);
}
```

```

long long sumQSort(long long* a, const long long middle,
                  const long long limiter)
{
    long long size = middle + 1;
    long long* qSortArray = (long long*)malloc(sizeof(long long) *
                                                size);

    arrayCopy(qSortArray, a, size);
    qsort(qSortArray, size, sizeof(long long),
          compareInts);
    reverseArray(qSortArray, size);

    long long sum = 0;
    for (size_t i = 0; i < size; i += 2)
    {
        sum += qSortArray[i];

        if (sum > limiter)
        {
            free(qSortArray);

            return limiter + 1;
        }
    }

    free(qSortArray);

    return sum;
}

long long binSearch(long long* a, const size_t n,
                   const long long limiter)
{
    long long left = -1;
    long long right = n;

    while (right - left > 1)
    {
        long long middle = (left + right) / 2;

        if (sumQSort(a, middle, limiter) > limiter)
            right = middle;
        else
            left = middle;
    }

    return right;
}

int main(void)
{
    long long n, h;
    scanf("%lld %lld", &n, &h);

    long long* a = (long long*)malloc(sizeof(long long) *
                                        n);

    inputArray(a, n);

    long long answer = binSearch(a, n, h);

    printf("%lld", answer);

    free(a);

    return 0;
}

```


Вердикт тестирующей системы:

180285767	10.11.2022 12:58	Saryi_Sky	1119В - Алена и узкий холодильник	GNU C11	Полное решение	15 мс	0 КБ
---------------------------	---------------------	-----------	---	------------	----------------	-------	------

Задача №12. Модные числа.

Код программы:

[illegible]

```

bool isTwoTreangularNumBinSearch
(const long long* const a, const long long size,
 const long long n)
{
    for (long long i = size - 2; i >= 0; i--)
    {
        long long secondNum = n - a[i];

        long long left = -1;
        long long right = i + 1;

        while (right - left > 1)
        {
            long long middle = (left + right) / 2;

            if (a[middle] > secondNum)
                right = middle;
            else
                left = middle;
        }

        if (a[left] == secondNum)
            return true;
    }

    return false;
}

int main(void)
{
    long long n;
    scanf("%lld", &n);

    long long* a;
    size_t sizeA;
    arrayOfArithmeticProgression(&a, n, &sizeA);

    printf(isTwoTreangularNumBinSearch(a, sizeA, n) ?
           "YES" : "NO");

    free(a);

    return 0;
}

```

Вердикт тестирующей системы:

180314584	10.11.2022 17:17	Staryi_Sky	192A - Модные числа	GNU C11	Полное решение	60 мс	1100 КБ
---------------------------	---------------------	----------------------------	-------------------------------------	------------	----------------	-------	---------

Задача №13. Пара тем.

Код программы:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>
#include <stdlib.h>

void inputArray(long long* a, const size_t n)
{
    for (size_t i = 0; i < n; i++)
        scanf("%lld", &a[i]);
}

void completionDifferencesArray(long long* diff,
    size_t n, const long long* a, const long long* b)
{
    while (n--)
        *diff++ = (*a++) - (*b++);
}

int compareInts(const void* a, const void* b)
{
    int arg1 = *(const int*)a;
    int arg2 = *(const int*)b;

    if (arg1 < arg2)
        return -1;
    if (arg1 > arg2)
        return 1;

    return 0;
}

void swap(long long* a, long long* b)
{
    long long t = *a;
    *a = *b;
    *b = t;
}

void reverseArray(long long* a, const size_t n)
{
    for (size_t i = 0, j = n - 1; i < j; i++, j--)
        swap(&a[i], &a[j]);
}
```

```

long long lessonsCounter(const long long* const a,
                          const size_t n)
{
    long long counter = 0;

    for (size_t i = 0; i < n && a[i] > 0; i++)
    {
        long long left = i;
        long long right = n;

        while (right - left > 1)
        {
            long long middle = (right + left) / 2;

            if (-a[middle] >= a[i])
                right = middle;
            else
                left = middle;
        }

        counter += left - i;
    }

    return counter;
}

int main(void)
{
    long long n;
    scanf("%lld", &n);

    long long* a = (long long*)malloc
        (sizeof(long long) * n);
    inputArray(a, n);
    long long* b = (long long*)malloc
        (sizeof(long long) * n);
    inputArray(b, n);

    long long* differences = (long long*)malloc
        (sizeof(long long) * n);
    completionDifferencesArray(differences, n, a, b);
    qsort(differences, n, sizeof(long long),
        compareInts);
    reverseArray(differences, n);

    free(a);
    free(b);

    long long answer = lessonsCounter(differences, n);
    printf("%lld", answer);

    free(differences);

    return 0;
}

```

Вердикт тестирующей системы:

180397468	10.11.2022 23:21	Staryi_Sky	1324D - Пара тем	GNU C11	Полное решение	124 мс	4700 КБ
---------------------------	---------------------	------------	----------------------------------	------------	----------------	--------	---------

Вывод: в ходе выполнения работы получены навыки использования алгоритмов бинарного поиска для решения задач оптимизации.