Образец оформления отчёта по Лабораторной работе.

|  |  |
| --- | --- |
| **К Г Э У** | МИНИСТЕРСТВО ВЫСШЕГО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ ЭНЕРГЕТИЧЕСКИЙ УНИВЕРСИТЕТ»**  (ФГБОУ ВО «КГЭУ») |

**Кафедра Информатики и информационных управляющих систем**

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №9**   
**ДИНАМИЧЕСКОЕ ВЫДЕЛЕНИЕ ПАМЯТИ. РАБОТА СО ВЛОЖЕННЫМИ ЦИКЛАМИ.**

|  |  |
| --- | --- |
| **Исполнитель:** | Соловьёв Леонид |
| **Группа:** | ПИ-1-22 |
|  |  |
|  |  |

Казань – 2022

1.1  
1.1  
  
Код  
#include<iostream>

#include <cmath>

double series(double x, int N);

void sum\_dont\_grow\_index\_or\_max\_sum(double \*arr, int n);

void max\_sum(double \*arr, int n);

int main(){

    double \*mass\_sum = new double[5];

    double x;

    std::cout << "Input x: "; std::cin >> x;

    for (int i = 5; i <= 25; i+=5) {

        mass\_sum[(i-5)/5] = series(x, i);

        std::cout << "Sum" << i << " = " << mass\_sum[(i-5)/5] << '\n';

    }

    sum\_dont\_grow\_index\_or\_max\_sum(mass\_sum, 5);

    return 0;

}

double series(double x, int N) {

    double sum = 0;

    double U = x \* x;

    for (int i = 1; i <= N; i++) {

        U = ((U / x) + (1. / pow(x, i)));

        sum += U;

    }

    return sum;

}

void sum\_dont\_grow\_index\_or\_max\_sum(double \*arr, int n) {

    bool flag = true;

    for (int i = 0; i < (n-1); i++) {

        if (arr[i+1]/arr[i] <= 1.1) {

            flag = false;

            std::cout << "N, where sum don't grow greater than ten percent is " << (i+1) \* 5;

            return;

        }

    }

    if (flag) {

        max\_sum(arr, n);

    }

}

void max\_sum(double \*arr, int n) {

    double max = -10000;

    int ind = -1;

    for (int i = 0; i < n; i++) {

        if (arr[i] > max) {

            max = arr[i];

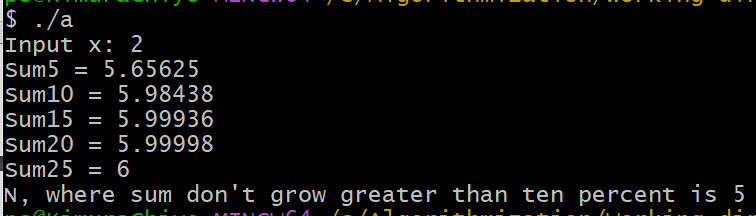
            ind = i;

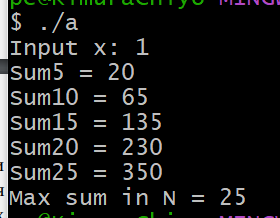
        }

    }

    std::cout << "Max sum in N = " << (ind+1)\*5;

}  
Решение





1.2  
  
Код

#include<iostream>

#include <cmath>

double series(double x, int N);

void sum\_dont\_grow\_index\_or\_max\_sum(double \*arr, int n);

void max\_sum(double \*arr, int n);

int main(){

    double \*mass\_sum = new double[5];

    double x;

    std::cout << "Input x: "; std::cin >> x;

    for (int i = 5; i <= 25; i+=5) {

        mass\_sum[(i-5)/5] = series(x, i);

        std::cout << "Sum" << i << " = " << mass\_sum[(i-5)/5] << '\n';

    }

    sum\_dont\_grow\_index\_or\_max\_sum(mass\_sum, 5);

    return 0;

}

double series(double x, int N) {

    double sum = 0;

    float k;

    double U = log(3 \* x + 1) \* pow(x, 3) / 1;

    for (int i = 1; i <= N; i++) {

        sum += U;

        k = log(3 \* x + i + 1) / log(3 \* x + i);

        U \*= (k \* pow(x, 3) / (i + 1));

    }

    return sum;

}

void sum\_dont\_grow\_index\_or\_max\_sum(double \*arr, int n) {

    bool flag = true;

    for (int i = 0; i < (n-1); i++) {

        if (arr[i+1]/arr[i] <= 1.1) {

            flag = false;

            std::cout << "N, where sum don't grow greater than ten percent is " << (i+1) \* 5;

            return;

        }

    }

    if (flag) {

        max\_sum(arr, n);

    }

}

void max\_sum(double \*arr, int n) {

    double max = -10000;

    int ind = -1;

    for (int i = 0; i < n; i++) {

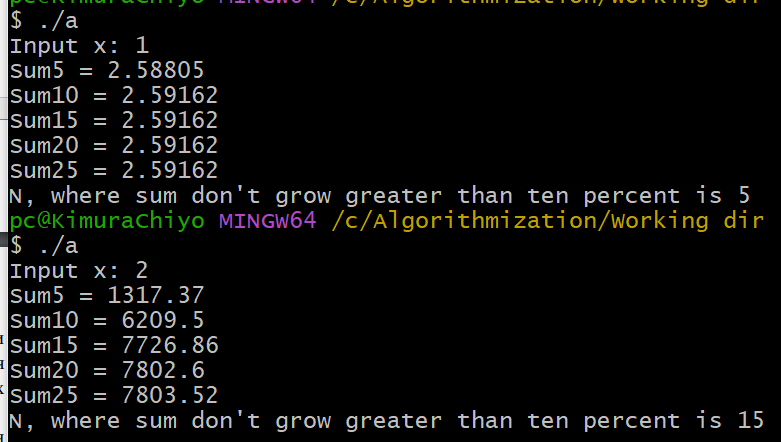
        if (arr[i] > max) {

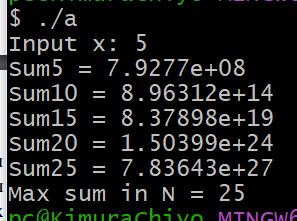
            max = arr[i];

            ind = i;

        }

    }

    std::cout << "Max sum in N = " << (ind+1)\*5;  
}  
Решение  




2.1  
  
Код

#include<iostream>

#include <cmath>

#include <string>

#include <ctime>

void init\_mass(int \*\*arr, int m, int n);

std::string max\_month(int \*\*arr, int m, int n);

int min\_raion(int \*\*arr, int m, int n);

int main(){

    srand(time(NULL));

    int n;

    std::cout << "How many raion in your city? "; std::cin >> n;

    int \*\*mass\_amount = new int \* [n], x;

    for (int i=0; i < n; i++) {

        mass\_amount[i] = new int [12];

    }

    init\_mass(mass\_amount, n, 12);

    std::string S = max\_month(mass\_amount, n, 12);

    std::cout << "\nMonth with max amount of people is " << S << "\n\n";

    int min\_amount =  min\_raion(mass\_amount, n, 12);

    std::cout << "\nMin amount people in Raion" << min\_amount+1;

    return 0;

}

void init\_mass(int \*\*arr, int m, int n) {

    for (int i = 0; i < m; i++) {

        std::cout << "Raion" << i+1 << ":\n";

        std::string month = "";

        for (int k = 0; k < n; k++) {

            arr[i][k] = rand()%1501 + 500;

            switch (k) {

                case 0: { month = "January"; break;}

                case 1: { month = "February"; break;}

                case 2: { month = "March"; break;}

                case 3: { month = "April"; break;}

                case 4: { month = "May"; break;}

                case 5: { month = "June"; break;}

                case 6: { month = "July"; break;}

                case 7: { month = "August"; break;}

                case 8: { month = "September"; break;}

                case 9: { month = "October"; break;}

                case 10: { month = "November"; break;}

                case 11: { month = "December"; break;}

            }

        std::cout << "\tMonth: " << month << "\b\t\tAmount: " << arr[i][k] << '\n';

        }

        std::cout << '\n';

    }

}

std::string max\_month(int \*\*arr, int m, int n) {

    std::string month;

    int ind\_month;

    int sum = 0;

    int max\_sum = 0;

    for (int i = 0; i < n; i++) {

        for (int k = 0; k < m; k++) {

            sum += arr[k][i];

        }

        switch (i) {

            case 0: { month = "January"; break;}

            case 1: { month = "February"; break;}

            case 2: { month = "March"; break;}

            case 3: { month = "April"; break;}

            case 4: { month = "May"; break;}

            case 5: { month = "June"; break;}

            case 6: { month = "July"; break;}

            case 7: { month = "August"; break;}

            case 8: { month = "September"; break;}

            case 9: { month = "October"; break;}

            case 10: { month = "November"; break;}

            case 11: { month = "December"; break;}

        }

        std::cout << "General Amount of " << month << ": " << sum << '\n';

        if (sum > max\_sum) {

            ind\_month = i;

            max\_sum = sum;

        }

        sum = 0;

    }

    switch (ind\_month) {

        case 0: { month = "January"; break;}

        case 1: { month = "February"; break;}

        case 2: { month = "March"; break;}

        case 3: { month = "April"; break;}

        case 4: { month = "May"; break;}

        case 5: { month = "June"; break;}

        case 6: { month = "July"; break;}

        case 7: { month = "August"; break;}

        case 8: { month = "September"; break;}

        case 9: { month = "October"; break;}

        case 10: { month = "November"; break;}

        case 11: { month = "December"; break;}

    }

    return month;

}

int min\_raion(int \*\*arr, int m, int n) {

    int ind\_raion;

    int sum = 0;

    int min\_sum = 1000000;

    for (int i = 0; i < m; i++) {

        for (int k = 0; k < n; k++) {

            sum += arr[i][k];

        }

        std::cout << "General Amount People in Raion" << i+1 << ": " << sum << std::endl;

        if (sum < min\_sum) {

            ind\_raion = i;

            min\_sum = sum;

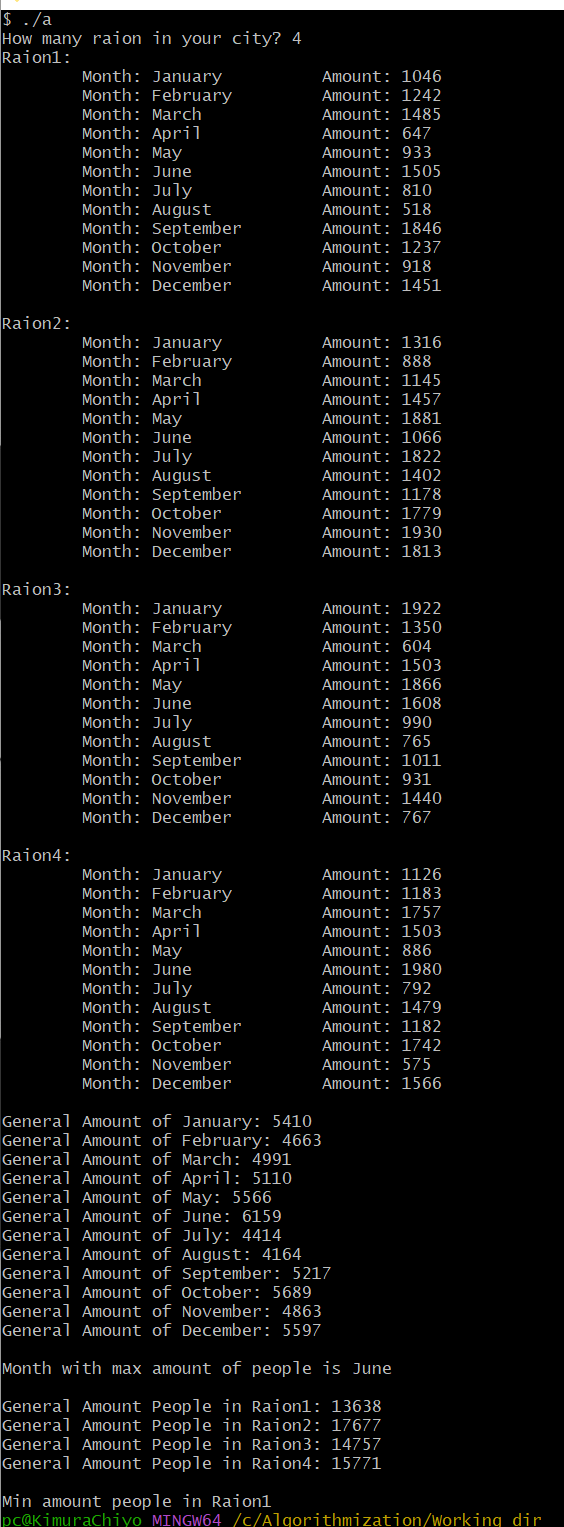
        }

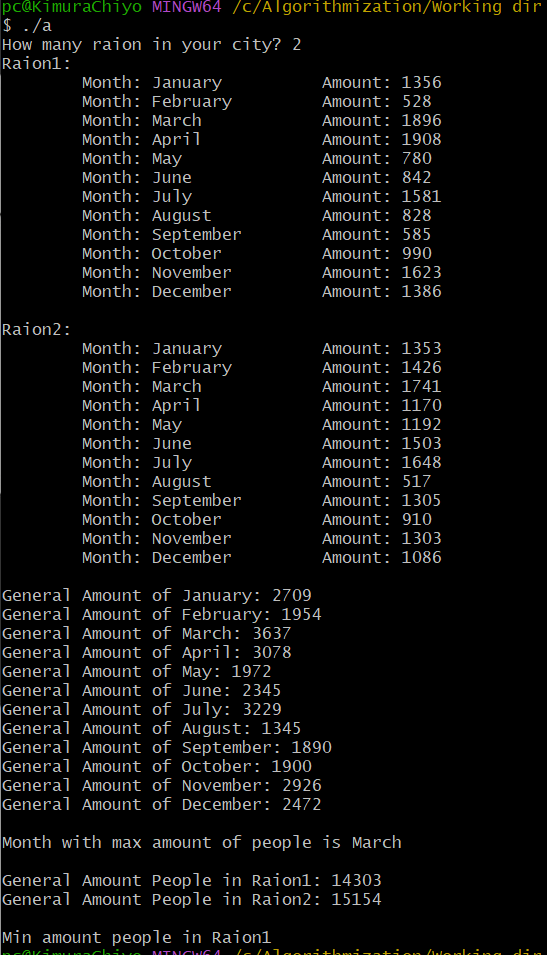
        sum = 0;

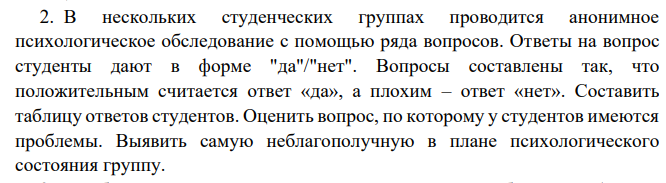
    }

    return ind\_raion;

}

Решение  




2.2  
  
Код  
#include<iostream>

#include <cmath>

#include <string>

#include <ctime>

void init\_mass(int \*\*arr, int m, int n, int x);

void max\_count\_no\_answer(int \*\*arr, int n, int m, int x);

void max\_count\_no\_group(int \*\*arr, int n, int m, int x);

int main(){

    srand(time(NULL));

    int n;

    std::cout << "How many groups in test? "; std::cin >> n;

    int x;

    std::cout << "How many peoples in group? "; std::cin >> x;

    int m;

    std::cout << "How many questions in test? "; std::cin >> m;

    int \*\*mass\_amount = new int \* [n];

    for (int i=0; i < n; i++) {

        mass\_amount[i] = new int [m];

    }

    init\_mass(mass\_amount, n, m, x);

    max\_count\_no\_answer(mass\_amount, n, m, x);

    std::cout << '\n';

    max\_count\_no\_group(mass\_amount, n, m, x);

    return 0;

}

void init\_mass(int \*\*arr, int n, int m, int x) {

    for (int i = 0; i < n; i++) {

        std::cout << "Group" << i+1 << ":\n";

        std::string month = "";

        for (int k = 0; k < m; k++) {

            arr[i][k] = rand()%(x+1);

            std::cout << "\tQuestion" << k+1 << "\b\tAmount of yes: " << arr[i][k] << "\tAmount of no " << x - arr[i][k] << '\n';

        }

        std::cout << '\n';

    }

}

void max\_count\_no\_answer(int \*\*arr, int n, int m, int x) {

    int max\_no\_sum = 0;

    int ans;

    int sum = 0;

    for (int i = 0; i < m; i++) {

        for (int k = 0; k < n; k++) {

            sum += arr[k][i];

        }

        std::cout << "Question" << i+1 << ":\n\tGeneral Amount of yes: " << sum << "\n\tGeneral Amount of no: " << x\*n - sum << std::endl;

        if (x\*n - sum > max\_no\_sum) {

            max\_no\_sum = (x\*n - sum);

            ans = i;

        }

        sum = 0;

    }

    std::cout << "\nQuestion with bad rating is Question" << ans+1 << std::endl;

}

void max\_count\_no\_group(int \*\*arr, int n, int m, int x) {

    int max\_no\_sum = 0;

    int ans;

    int sum = 0;

    for (int i = 0; i < n; i++) {

        for (int k = 0; k < m; k++) {

            sum += arr[i][k];

        }

        std::cout << "Group" << i+1 << ":\n\tGeneral Amount of yes: " << sum << "\n\tGeneral Amount of no: " << x\*m - sum << std::endl;

        if (x\*m - sum > max\_no\_sum) {

            max\_no\_sum = (x\*m - sum);

            ans = i;

        }

        sum = 0;

    }

    std::cout << "\nGroup with bad rating is Group" << ans+1;

}

Решение  
