

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Вятский государственный университет»
Факультет автоматики и вычислительной техники
Кафедра электронных вычислительных машин

Лабораторная работа № 1 по курсу
«Технологии программирования»

Выполнил студент группы ИВТ-21 _____/Рзаев А. Э./
Проверил доцент кафедры ЭВМ _____/Долженкова М. Л./

Киров 2017

1 Задание

Написать программу, реализующую вычисление функций $\exp x$, $\cos x$ и $\sin x$, принимающих в качестве аргумента числа типов `int` и `double`, используя перегруженные функции и шаблоны.

2 Листинг программы

Листинг разработанной программы приведен в приложении А.

3 Экранные формы

Экранные формы приведены в приложении Б.

4 Вывод

В ходе выполнения лабораторной работы были изучены основные принципы использования стандартных потоков ввода-вывода, синтаксис шаблонных и перегруженных функций.

Приложение А

(обязательное)

Листинг программы

```
#include <bits/stdc++.h> // "all in one" header file

template<class T>
double exp_template(T x, bool flag=false) {
    double res;
    if (!flag) {
        res = 1;
        double val = x;
        uint64_t fac = 1;
        for (uint64_t n = 1; n < 15; ++n, val *= x, fac *= n) {
            res += val / fac;
        }
    }
    else {
        res = exp_template(1.0);
        double val = res;
        for (uint64_t n = 1; n < x; ++n) {
            res *= val;
        }
    }
    return res;
}

// sin_overload
double sincos_overload(double x) {
    double res = 0;
    double val = x;
    int sign = 1;
    uint64_t fac = 1;
    uint64_t n = 1;
    while (n < 15) {
        if (n % 2 != 0) {
            res += val / fac * sign;
            sign *= -1;
        }
        ++n;
        fac *= n;
        val *= x;
    }
}
```

```

        return res;
    }

// sin_overload
double sincos_overload(int64_t x) {
    double res = 1;
    double val = static_cast<double>(x * x);
    int sign = -1;
    uint64_t fac = 2;
    uint64_t n = 2;
    while (n < 15) {
        res += val / fac * sign;

        n += 2;
        fac *= n * (n - 1);
        sign *= -1;
        val *= x * x;
    }

    return res;
}

void print_help(std::ostream& os) {
    os
        << "What do you want to do (enter number between 0 and 4):"
    << std::endl
        << "1. Compute exp(x) where x is a float point value" <<
    std::endl
        << "2. Compute exp(x) where x is an unsigned integer value"
    << std::endl
        << "3. Compute sin(x) where x is a float point value" <<
    std::endl
        << "4. Compute cos(x) where x is a signed integer value" <<
    std::endl
        << "0. Exit" << std::endl;
}

template<class T>
void print_result(std::ostream& os, const T& val) {
    os << "The result of computation is " << val << std::endl;
}

void print_error(std::ostream& os) {
    os << "Unable to obtain value" << std::endl;
}

bool try_get_value(std::istream& is, double& val) {
    std::string s;
    bool ok = true;

```

```

        is >> s;
        size_t err = 0;
        try {
            val = std::stod(s, &err);
        }
        catch (std::exception& ex) {
            ok = false;
        }

        ok = ok && !(err < s.size());
        std::getline(is, s);

        return ok;
    }

bool try_get_value(std::istream& is, uint64_t& val) {
    std::string s;
    bool ok = true;

    is >> s; if (s.size() > 0 && s[0] == '-1') s = false;
    size_t err = 0;
    try {
        val = std::stoull(s, &err);
    }
    catch (std::exception& ex) {
        ok = false;
    }

    ok = ok && !(err < s.size());
    std::getline(is, s);

    return ok;
}

bool try_get_value(std::istream& is, int64_t& val) {
    std::string s;
    bool ok = true;

    is >> s;
    size_t err = 0;
    try {
        val = std::stoll(s, &err);
    }
    catch (std::exception& ex) {
        ok = false;
    }

    ok = ok && !(err < s.size());
    std::getline(is, s);

```

```

        return ok;
    }

int main(void) {
    using std::cin;
    using std::cout;

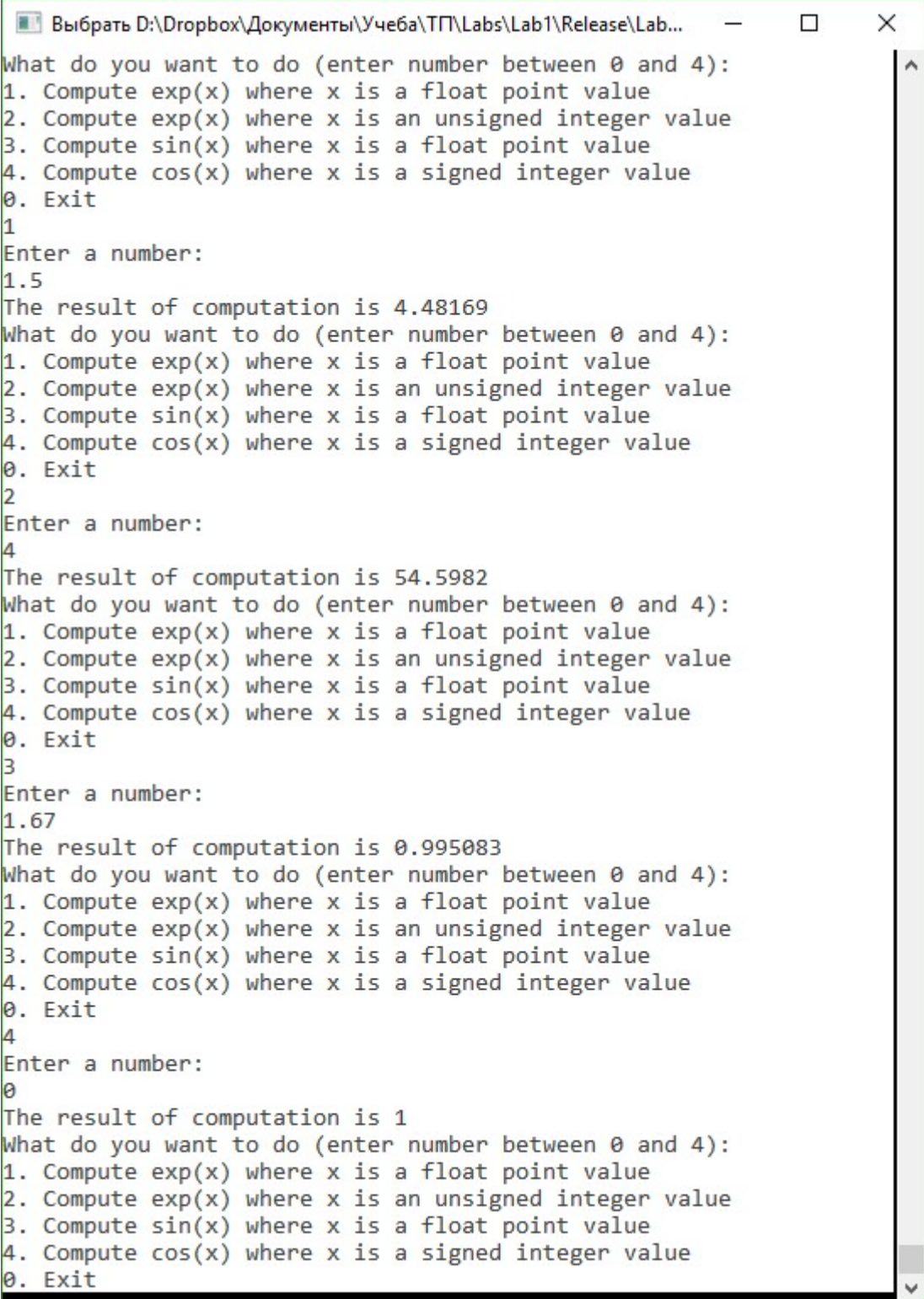
    while (true) {
        print_help(cout);
        int64_t n = 0;
        if (!try_get_value(cin, n) || !(n >= 0 && n < 5)) {
            cout << "Invalid code of operation" << std::endl;
            continue;
        }
        if (n == 0)
            break;

        cout << "Enter a number: " << std::endl;

        if (n == 1) {
            double val;
            if (try_get_value(cin, val))
                print_result(cout, exp_template(val));
            else
                print_error(cout);
        }
        else if (n == 2) {
            uint64_t val;
            if (try_get_value(cin, val))
                print_result(cout, exp_template(val, true));
            else
                print_error(cout);
        }
        else if (n == 3) {
            double val;
            if (try_get_value(cin, val))
                print_result(cout, sincos_overload(val));
            else
                print_error(cout);
        }
        else if (n == 4) {
            int64_t val;
            if (try_get_value(cin, val))
                print_result(cout, sincos_overload(val));
            else
                print_error(cout);
        }
    }
    return 0;
}

```

Приложение Б
(обязательное)
Экранные формы



```
Выбрать D:\Dropbox\Документы\Учеба\ТП\Лabs\Lab1\Release\Lab...
What do you want to do (enter number between 0 and 4):
1. Compute exp(x) where x is a float point value
2. Compute exp(x) where x is an unsigned integer value
3. Compute sin(x) where x is a float point value
4. Compute cos(x) where x is a signed integer value
0. Exit
1
Enter a number:
1.5
The result of computation is 4.48169
What do you want to do (enter number between 0 and 4):
1. Compute exp(x) where x is a float point value
2. Compute exp(x) where x is an unsigned integer value
3. Compute sin(x) where x is a float point value
4. Compute cos(x) where x is a signed integer value
0. Exit
2
Enter a number:
4
The result of computation is 54.5982
What do you want to do (enter number between 0 and 4):
1. Compute exp(x) where x is a float point value
2. Compute exp(x) where x is an unsigned integer value
3. Compute sin(x) where x is a float point value
4. Compute cos(x) where x is a signed integer value
0. Exit
3
Enter a number:
1.67
The result of computation is 0.995083
What do you want to do (enter number between 0 and 4):
1. Compute exp(x) where x is a float point value
2. Compute exp(x) where x is an unsigned integer value
3. Compute sin(x) where x is a float point value
4. Compute cos(x) where x is a signed integer value
0. Exit
4
Enter a number:
0
The result of computation is 1
What do you want to do (enter number between 0 and 4):
1. Compute exp(x) where x is a float point value
2. Compute exp(x) where x is an unsigned integer value
3. Compute sin(x) where x is a float point value
4. Compute cos(x) where x is a signed integer value
0. Exit
```

Рисунок Б.1 – Основное окно программы