МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Гжельский государственный университет»** (ГГУ)

Колледж ГГУ

Специальность 09.02.07 Информационные системы и программирования

**Отчёт по лабораторной работе №7**

**По дисциплине «МДК 02.02 Управление проектами»**

**на тему «Метрики Холстеда»**

ВЫПОЛНИЛА:

Студентка группы ИСП-О-17

Янгабозова Анастасия

ПРОВЕРИЛА:

Прокуронова А.Ю.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

п. Электроизолятор

2019 г.

**Цель работы:** Изучение основ метрической теории программы Холстеда, расчет количественных характеристик для индивидуального модуля.

Код:

#include "pch.h"

#include <iostream>

using namespace std;

int main()

{

cout.fill('\*');

cout.width(3);

cout << "";

cout.fill(' ');

cout.width(7);

cout << "";

cout.fill('\*');

cout.width(3);

cout << "";

cout << endl;

cout.fill(' ');

cout.width(1);

cout << "";

cout.fill('\*');

cout.width(3);

cout << "";

cout.fill(' ');

cout.width(5);

cout << "";

cout.fill('\*');

cout.width(3);

cout << "";

cout << endl;

}

* n1 - 3
* n2 – 4
* N1 - 11
* N2 - 25
* n2\*- 3
* N – 36
* n – 7
* S = 18
* T' =6882(сек)
* E0 = 11.61 x 11.61 x 11.61/(0.351 x 0.351) = 12702.302

**Длина программы**:

N'= n1 x log2(n1) + n2 x log2(n2)

N' = 3 log2(3) + 4 log2(4) = 12.76

**Объём программы:**

V = N \* log2(n)

V = 36 \* log2(7) = 101.1

**Потенциальный объём:**

V\* = (2+ n2\*) \* log2(2+ n2\*)

V\* = (2+ 3) \* log2(2+ 3) = 11.61

**граничный объем:**

V\*\* = (2+ (n2\*)\* log2(n2\*)) \* log2(2+ n2\*)

V\*\* = (2+ (3)\* log2(3)) \* log2(2+ 3) = 15.7

**соотношения между операциями и операндами:**

A = n2\* /(n2\* +2) \* log2(n2\*/2)

B = n2\* - 2 \* A

A = 3 /(3 +2) \* log2(3/2) = 0.351

B = 3 - 2 \* 0.351 = 2.298

**уровень программы, интеллектуальное содержание:**

L = V\*/V

L = 11.61/101.1 = 0.1115

**работа по программированию:**

E = V 2/ V\*

E = 101.1 х 2/ 11.61 = 17.42

**приближенное время программирования:**

T' = E/S

T' = 17.42/18 = 0.968

**уровень языка:**

A = L \* L\* V

A = 0.1115 \* 0.1115 \* 101.1 = 1.26

**число переданных ошибок в программе**

B = V/Е0

B = 101.1/12702.302 = 0.008

**Контрольные вопросы:**

**1. Где можно использовать метрики Холстеда?**

* оценки топологической и информационной сложности программ;
* оценки надежности программных систем;
* оценки производительности ПО;
* оценки уровня языковых средств и их применения;
* оценки возможности понимания программных текстов, что необходимо для сопровождения и модификации программ;
* оценки производительности труда программистов для составления планов и графиков работ по созданию ПО.

**2. Чем определяются характеристики программы?**

алгоритмическая сложность (логика алгоритмов обработки информации);

– состав и глубина проработки реализованных функций обработки;

– полнота и системность функций обработки;

– объем файлов программ;

– требования к операционной системе и техническим средствам обработки со стороны программного средства;

– объем дисковой памяти;

– размер оперативной памяти для запуска программ;

– тип процессора;

– версия операционной системы;

– наличие вычислительной сети и др

**3. Как оценить качество реализации алгоритма по метрикам?**

В проводимом экспериментальном исследовании вычислялись стандартные метрики для алгоритмов обнаружения объектов:

- пересечение, деленное на объединение. Это мера соответствия двух рамок, может принимать значения от 0 (полное несовпадение) до 1 (полное совпадение) или от 0 до 100 %. Это отношение площади пересечения двух рамок к площади их объединения;

 - полнота и точность

**4. В чем недостаток программометрии?**

В крупных программных средах время от времени появляются механизмы подсчета различных метрик. Волнообразный интерес к теме так выглядит потому, что до сих пор в метриках не придумано главного — что с ними делать.