1. **Лабораторная работа №1**

**«Асимптотический анализ количественно-зависимых алгоритмов»**

* 1. **Цель работы**

Изучить поведение функций трудоемкости количественно-зависимых алгоритмов в реальных интервалах значений мощности множества исходных данных. На основании этого сделать предпочтительный выбор того или иного алгоритма. Для сравнения функций трудоемкости использовать аппарат интервального анализа, реализованный в виде программы на языке С++.

* 1. **Вариант задания ­– 14**

Требуется для указанных по варианту пары функций трудоёмкости и , целочисленных интервалов {(20; 50), (100; 120), (500; 540)} и значений , определить, каково соотношение между функциями трудоемкости на заданном интервале. Для вычисления значений функций и определения соотношений написать программу на языке С. Результаты работы программы сохранять в текстовые файлы.

Путем подбора значений аргумента определить интервалы, на которых выполняется соотношение: .

Построить графики заданных функций на указанном интервале (вручную или с помощью любого программного обеспечения).

* 1. **Ход работы**
     1. **Текст программы**

В соответствии с вариантом была написана программа на языке Си и её текст представлен ниже.

#include <math.h>

#include <stdio.h>

int main() {

double Fn, //F(n)

Gn, //G(n)

ATg\_FG,

ATg\_GF,

pi, // ATg\_FG - ATg\_GF

Nbegin, //Левая граница интервала ->вводим с клавы

Nend, //Правая граница интервала ->вводим с клавы

step, //Шаг изменения аргумента ->вводим с клавы

phi, //Значение угла

k, //Коэффициент кратности ->вводим с клавы

Delta, //Оценка «Дельта»

Theta, //Оценка «Тетта»

O\_large, //Оценка «О-большое»

n; //Значение аргумента функций трудоемкости (n)

FILE \*stream; //Указатель на файл, в который осуществляется ввод-вывод расчетов

stream = fopen("Example\_TA.TXT", "w+"); // открытие файла для записи

// Ввод значений границ интервалов, шага изменения аргумента внутри интервала, коэффициента кратности

printf("Input Nbegin ->"); scanf("%lf", &Nbegin); //Левая граница, ввод значения

printf("Input Nend ->"); scanf("%lf", &Nend); //Правая граница, ввод значения

printf("Input step ->"); scanf("%lf", &step); //Шаг изменения аргумента

printf("Input koefficient ->"); scanf("%lf", &k); //Коэффициент кратности

phi = M\_PI/k; // Определили угол изменения как П/k, (M\_PI - встроенная константа языка Си = числу Pi)

n = Nbegin; //Аргумент функций равен левой границе интервала

fprintf(stream, "n, F(n), G(n), ATg\_FG, ATg\_GF, pi, Delta, Theta, O\_large \n\n");

while (n <= Nend) {

Fn = (powl(2, (n-4))) / sqrtl(n); //Расчет значения функции F(n)

Gn = powl(1.618, n); //Расчет значения функции G(n)

ATg\_FG = atan(Fn/Gn);

ATg\_GF = atan(Gn/Fn);

pi = ATg\_FG - ATg\_GF;

Delta = phi - pi; //Оценка «Дельта»

Theta = fabs(pi) - phi; //Оценка «Тетта»

O\_large = pi + phi; //Оценка «О-большое»

//Запись расчетов в файл

fprintf(stream, "%lf %lf \t %lf\t %lf %lf %lf %lf %lf %lf\n",

n, Fn, Gn, ATg\_FG, ATg\_GF, pi, Delta, Theta, O\_large);

n = n + step; //Получение следующего значения аргумента

}

fclose(stream); //Закрыли файл

}

* + 1. **Анализ результатов работы программы**

Составленная программа была скомпилирована и выполнена для каждого из предложенных интервалов. Таблица 1.1 содержит результат выполнения программы для интервала (30; 47.5) с шагом 0.5 и k = 32.

Таблица 1.1 – Результат выполнения программы

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| n | F(n) | G(n) | ATg\_FG | ATg\_GF | pi | Delta | Theta | O\_large |
| 30 | 12252346.207115 | 1859325.89710700 | 1.420193 | 0.150604 | 1.269589 | -1.171414 | 1.171414 | 1.367764 |
| 30.5 | 17184819.124802 | 2365074.234965 | 1.43403 | 0.136767 | 1.297263 | -1.199088 | 1.199088 | 1.395438 |
| 31 | 24106215.574788 | 3008389.30152 | 1.446641 | 0.124155 | 1.322486 | -1.224311 | 1.224311 | 1.42066 |
| 31.5 | 33819688.5546770 | 3826690.112174 | 1.458126 | 0.112671 | 1.345455 | -1.24728 | 1.24728 | 1.44363 |
| 32 | 47453132.8121260 | 4867573.8 | 1.468577 | 0.102219 | 1.366358 | -1.268184 | 1.268184 | 1.464533 |
| 32.5 | 66590641.0785860 | 6191584.60149700 | 1.478083 | 0.092713 | 1.38537 | -1.287195 | 1.287195 | 1.483545 |
| 33 | 93457229.9117580 | 7875734.55379100 | 1.486724 | 0.084072 | 1.402652 | -1.304477 | 1.304477 | 1.500826 |
| 33.5 | 131178442.020206 | 10017983.8852220 | 1.494575 | 0.076221 | 1.418354 | -1.320179 | 1.320179 | 1.516529 |
| 34 | 184145203.624337 | 12742938.5080350 | 1.501706 | 0.06909 | 1.432616 | -1.334441 | 1.334441 | 1.53079 |
| 34.5 | 258526650.952220 | 16209097.9262890 | 1.50818 | 0.062616 | 1.445564 | -1.34739 | 1.34739 | 1.543739 |
| 35 | 362990988.413423 | 20618074.5060000 | 1.514057 | 0.05674 | 1.457317 | -1.359142 | 1.359142 | 1.555492 |
| 35.5 | 509718840.943525 | 26226320.4447360 | 1.519389 | 0.051407 | 1.467982 | -1.369807 | 1.369807 | 1.566157 |
| 36 | 715827882.666667 | 33360044.5507080 | 1.524227 | 0.04657 | 1.477657 | -1.379482 | 1.379482 | 1.575832 |
| 36.5 | 1005375798.98741 | 42434186.4795830 | 1.528614 | 0.042182 | 1.486432 | -1.388257 | 1.388257 | 1.584607 |
| 37 | 1412176547.94406 | 53976552.0830450 | 1.532593 | 0.038204 | 1.494389 | -1.396214 | 1.396214 | 1.592564 |
| 37.5 | 1983760419.60499 | 68658513.7239650 | 1.5362 | 0.034596 | 1.501603 | -1.403429 | 1.403429 | 1.599778 |
| 38 | 2786942793.86907 | 87334061.2703670 | 1.53947 | 0.031327 | 1.508143 | -1.409968 | 1.409968 | 1.606318 |
| 38.5 | 3915655591.41962 | 111089475.205375 | 1.542433 | 0.028363 | 1.51407 | -1.415896 | 1.415896 | 1.612245 |
| 39 | 5501961470.09366 | 141306511.135454 | 1.545119 | 0.025677 | 1.519442 | -1.421267 | 1.421267 | 1.617617 |
| 39.5 | 7731545183.36410 | 179742770.882296 | 1.547553 | 0.023244 | 1.524309 | -1.426134 | 1.426134 | 1.622484 |
| 40 | 10865503305.0357 | 228633935.017165 | 1.549757 | 0.021039 | 1.528718 | -1.430543 | 1.430543 | 1.626893 |
| 40.5 | 15270994830.2222 | 290823803.287556 | 1.551754 | 0.019042 | 1.532713 | -1.434538 | 1.434538 | 1.630887 |
| 41 | 21464358393.7553 | 369929706.857773 | 1.553563 | 0.017233 | 1.536331 | -1.438156 | 1.438156 | 1.634505 |
| 41.5 | 30171770043.2231 | 470552913.719265 | 1.555202 | 0.015595 | 1.539607 | -1.441432 | 1.441432 | 1.637782 |
| 42 | 42414581872.0274 | 598546265.695876 | 1.556685 | 0.014111 | 1.542575 | -1.4444 | 1.4444 | 1.640749 |
| 42.5 | 59629390197.5959 | 761354614.397771 | 1.558029 | 0.012767 | 1.545261 | -1.447087 | 1.447087 | 1.643436 |
| 43 | 83836975649.9575 | 968447857.895928 | 1.559245 | 0.011551 | 1.547694 | -1.449519 | 1.449519 | 1.645869 |
| 43.5 | 117880019833.856 | 1231871766.09559 | 1.560346 | 0.01045 | 1.549897 | -1.451722 | 1.451722 | 1.648071 |
| 44 | 165757614634.814 | 1566948634.07561 | 1.561343 | 0.009453 | 1.55189 | -1.453716 | 1.453716 | 1.650065 |
| 44.5 | 233095998896.980 | 1993168517.54267 | 1.562246 | 0.008551 | 1.553695 | -1.45552 | 1.45552 | 1.65187 |
| 45 | 327811032234.478 | 2535322889.93434 | 1.563062 | 0.007734 | 1.555328 | -1.457154 | 1.457154 | 1.653503 |
| 45.5 | 461040546878.806 | 3224946661.38404 | 1.563802 | 0.006995 | 1.556807 | -1.458632 | 1.458632 | 1.654981 |
| 46 | 648456580857.394 | 4102152435.91376 | 1.56447 | 0.006326 | 1.558144 | -1.45997 | 1.45997 | 1.656319 |
| 46.5 | 912112356638.505 | 5217963698.11938 | 1.565076 | 0.005721 | 1.559355 | -1.46118 | 1.46118 | 1.65753 |
| 47 | 1283042033899.06 | 6637282641.30847 | 1.565623 | 0.005173 | 1.56045 | -1.462275 | 1.462275 | 1.658625 |
| 47.5 | 1804920204742.54 | 8442665263.55715 | 1.566119 | 0.004678 | 1.561441 | -1.463266 | 1.463266 | 1.659616 |

На рисунке представлен график значений функций f(n) и g(n) и значений n из интервала (30; 47.5).

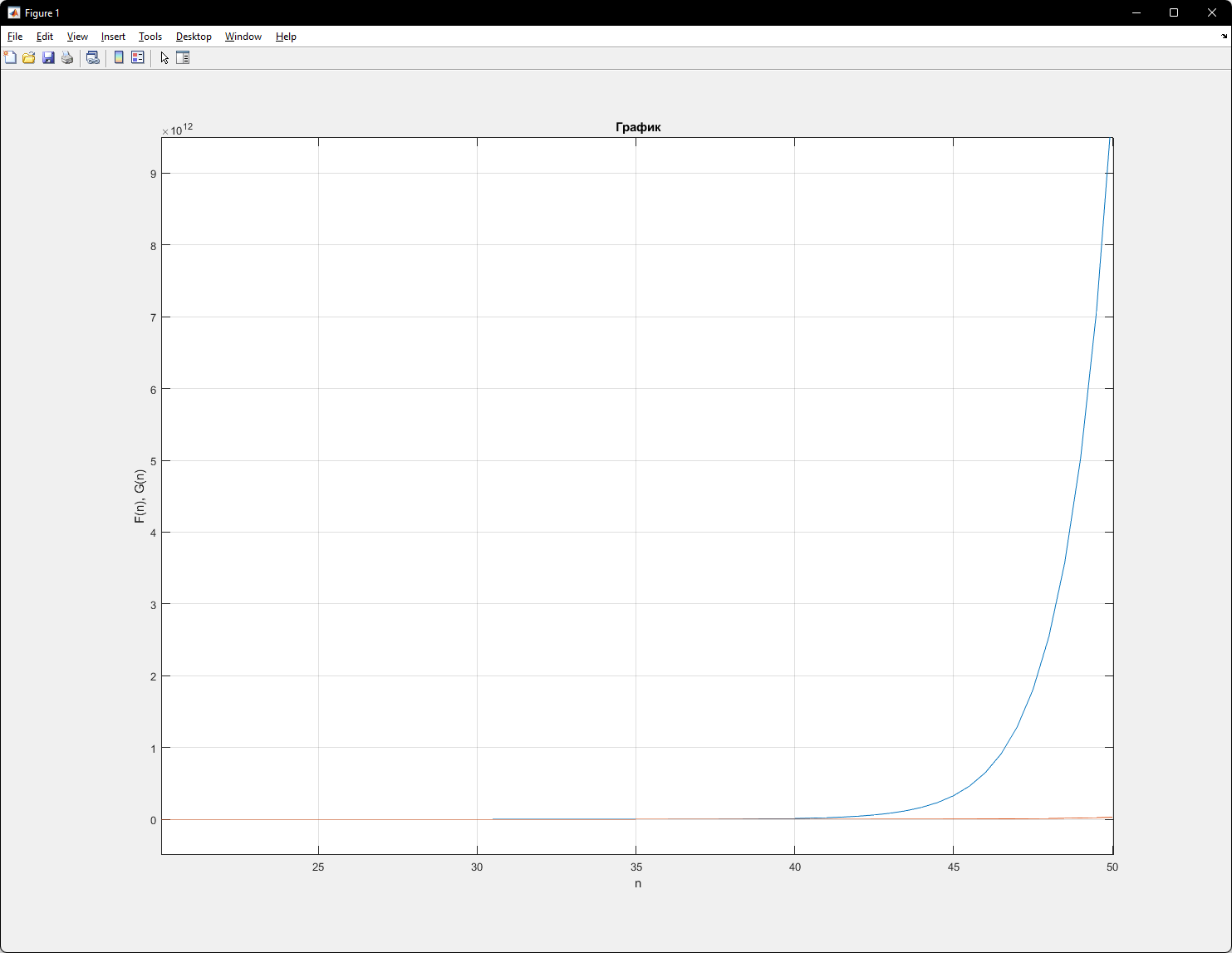


Рисунок 1.1 – График для интервала (30; 47.5)

Вывод

В ходе лабораторной работы было изучено поведение функций трудоемкости количественно-зависимых алгоритмов в реальных интервалах значений мощности множества исходных данных. Для сравнения функций трудоемкости использован аппарат интервального анализа, реализованный в виде программы на языке Cи. Для данных функций f(x) и g(x) интервал, на котором выполняется соотношение [30; 47.5], был построен график функций для этого интервала.