ГУАП

КАФЕДРА № 34

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  | Жиданов К.А. |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1 |
|  |
| по курсу: |
| ЯЗЫКИ ПРОГРАММИРОВАНИЯ |
|  |

РАБОТУ ВЫПОЛНИЛА

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТКА ГР. | 3145 |  |  |  | А.С. Дмитриева |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

**Вариант №6**

Вычислить значение корня квадратного из двух с заданной точностью

**Цель работы**

Изучить методы разработки консольных приложений, способы их запуска и обработки кодов возврата.

**Ход работы**

1. Реализуем на языке Си функцию, вычисляющую корень квадратный из двух. Данная функция будет вычислять корень с помощью вавилонского метода по формуле:

Код функции:

|  |
| --- |
| double nsqrt(double e) {  double n;  n = 1;  while (nfabs(2 - n\*n) > e) {  n = n / 2 + 1 / n;  }  return n;  } |

1. Составляем набор эталонных параметров и результатов для проверки работы программы. В качестве параметра будет выступать точность вычисления корня квадратного.

|  |  |
| --- | --- |
| Параметр | Результат |
| 0.01 | 1.41 |
| 0.001 | 1.414 |
| 0.00001 | 1.41421 |
| 0.000001 | 1.414213 |

1. Реализуем тестирующую функцию

|  |
| --- |
| double test\_sqrt() {  int r = 0;  r = r || (nfabs(nsqrt(0.1) - 1.5) >= 0.1);  r = r || (nfabs(nsqrt(0.001) - 1.414) >= 0.001);  r = r || (nfabs(nsqrt(0.00001) - 1.41421) >= 0.00001);  r = r || (nfabs(nsqrt(0.01) - 1.41) >= 0.01);  r = r || (nfabs(nsqrt(0.000001) - 1.414213) >= 0.000001);  return r;  } |

Для корректной работы данной функции напишем вспомогательную функцию fabs, возвращающую модуль числа:

|  |
| --- |
| double nfabs(double x) {  if (x > 0)  return x;  else  return -1 \* x;  } |

1. Реализуем программу, возвращающую код ошибки, соответствующий результату работы тестирующий функции:

|  |
| --- |
| double nfabs(double x) {  if (x > 0)  return x;  else  return -1 \* x;  }  double nsqrt(double e) {  double n;  n = 1;  while(nfabs(2 - n\*n) > e){  n = n / 2 + 1 / n;  }  return n;  }  double test\_sqrt() {  int r = 0;  r = r || (nfabs(nsqrt(0.001) - 1.414) >= 0.001);  r = r || (nfabs(nsqrt(0.00001) - 1.41421) >= 0.00001);  r = r || (nfabs(nsqrt(0.01) - 1.41) >= 0.01);  r = r || (nfabs(nsqrt(0.000001) - 1.414213) >= 0.000001);  return r;  }    int main(int argc, char\*\* argv) {  return test\_sqrt();  }; |

1. Компилируем программу, при имеющемся эталонном наборе данных программа работает корректно.

При добавлении неправильных данных в набор эталонных данных программа выдаёт код ошибки 1.

