**Санкт-Петербургский национальный исследовательский университет**

**информационных технологий, механики и оптики ** **УЧЕБНЫЙ ЦЕНТР ОБЩЕЙ ФИЗИКИ ФТФ**

Группа М3202 К работе допущен Студент Войнов Лев Витальевич, Чечулин Лев Работа выполнена Преподаватель Тимофеева Эльвира Олеговна Отчет принят

Отчет по моделированию №2

Интерференция волн

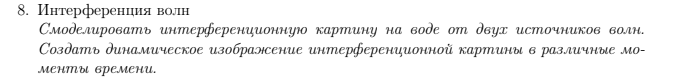
1. Цель работы.

Освоить навык комплексного решения физических и инженерных задач, используя методы численного моделирования процессов.

1. Задачи, решаемые при выполнении работы.

Написание программы для моделирования физического процесса

Удобная визуализация результата



1. Рабочие формулы.
2. Ход работы

Для каждой точки вычислим относительное значение высоты в конкретный момент

(от -1 до 1).

Нам нужно это нарисовать, значит необходимо положительное значение, для этого прибавим 1 и разделим на 2.

Далее выведем картинку в градации серого и сразу запустим подсчёт новой картинки.

С каждой новой картинкой время инкрементируется, а картинки выводятся в момент своей готовности, закрывая предыдущую. Из-за чего получается динамическое изображение волн на воде.

Программный код

Файл main.py (нужна библиотека pillow для запуска)

import math  
from PIL import Image  
  
  
class waveSource:  
 def \_\_init\_\_(self, x, y, wavelength, wavefrequency, amplitude, phase):  
 self.x = x  
 self.y = y  
 self.wavelength = wavelength  
 self.wavefrequency = wavefrequency  
 self.amplitude = amplitude  
 self.phase = phase  
  
  
class point:  
 def \_\_init\_\_(self, x, y):  
 self.x = x  
 self.y = y  
  
  
def distance(P, S):  
 return ((P.x-S.x)\*(P.x-S.x) + (P.y-S.y)\*(P.y-S.y)) \*\* 0.5  
  
  
WaveSourceArray = []  
for i in range(0, 2):  
 print("Wave source", i+1, ":")  
 print("Please, enter coordinate\_x of wave source in pixels (0 to 511): ")  
 x = int(input())  
 print("Please, enter coordinate\_y of wave source in pixels (0 to 511): ")  
 y = int(input())  
 print("Please, enter length of wave source in pixels: ")  
 wavelength = float(input())  
 print("Please, enter frequency of wave source in hertz: ")  
 wavefrequency = float(input())  
 print("Please, enter amplitude of this wave source in conventional units: ")  
 amplitude = float(input())  
 WaveSourceArray.append(waveSource(x, y, wavelength, wavefrequency, amplitude, 0))  
print("Please, enter phase difference: ")  
phase = float(input())  
WaveSourceArray[0].phase = phase  
# objectSize - условный размер объекта, т.к. если объект это материальная точка, то амплитуда будет бесконечной  
objectSize = 20  
v1 = WaveSourceArray[0].wavelength \* WaveSourceArray[0].wavefrequency  
v2 = WaveSourceArray[1].wavelength \* WaveSourceArray[1].wavefrequency  
A1 = WaveSourceArray[0].amplitude  
A2 = WaveSourceArray[1].amplitude  
b = distance(WaveSourceArray[0], WaveSourceArray[1])  
amplitudeMax = max(A1 / objectSize, A2 / objectSize, A1 / (v1/(v1+v2)\*b + objectSize) + A2 / (v2/(v1+v2)\*b + objectSize))  
image\_size = 128  
Arr = [[0]\*image\_size]\*image\_size  
for time in range (0, 100):  
 img = Image.new('RGB', (image\_size, image\_size))  
 for i in range (0, image\_size):  
 for j in range (0, image\_size):  
 Arr[i][j] = 0  
 for source in WaveSourceArray:  
 P = point(i, j)  
 x = distance(P, source)  
 l = source.wavelength  
 v = source.wavefrequency  
 A = source.amplitude  
 w = v\*2\*math.pi  
 k = 2\*math.pi/l  
 f = source.phase  
 if v\*l\*time >= x:  
 Arr[i][j] += A/(x+objectSize)\*math.cos(w\*time - x\*k + f)  
 color = int(round((Arr[i][j] + amplitudeMax) / 2 / amplitudeMax \* 255))  
 new\_color = (color, color, color)  
 img.putpixel((i, j), new\_color)  
 print(time)  
 img.show()  
 img.close()

Параметры для запуска вводятся прямо в консоль по запросу.