МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КІБЕРБЕЗПЕКИ

ЗВІТ ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ №1

із дисципліни «Алгоритми та структури даних»

за темою: «Структура даних стек, та основні операції»

Виконала студент групи Ін-13

Шеліхов Д.Ю.

Варіант 20

Суми – 2022

ЛАБОРАТОРНА РОБОТА №1

**Мета:** Навчитися використовувати структуру даних стек для розв’язанні завдань.

Завдання:

1. Реалізувати структуру даних "стек", застосувавши всі зазначені тут методи. Написати програму (функцію main), що містить опис стека яка моделює роботу стека.

Розмір стека (S) визначити наступним способом: S = варіант\_студента \* 5 + 50;

1. Заповнити стек випадково згенерували числами від 1 до 1000.
2. Створити 2 нових стека і перенести всі парні числа в 1-й стек, всі непарні - в 2-й.
3. Вивести в консоль елементи парного і непарного стеків.

Виконання:

***Стеком*** *(англ. Stack)* називається динамічна лінійна структура даних, додавання компоненти в яку і виключення компоненти з якої виробляється з одного кінця, званого вершиною стека. Стек працює за принципом LIFO (Last - In, First - Out) - надійшов останнім, обслуговується першим.

Я обрав мову програмування Python.

import random

import math

VARIANT = 20

class Steck():

def \_\_init\_\_(self,a = math.inf):

self.stack = list()

self.max\_size = a

def empty(self):

if len(self.stack)==0:

return 1

else:

return 0

def push(self,a):

if len(self.stack)>self.max\_size:

print("Превышен максимум")

else:

self.stack.append(a)

def pop(self):

if self.empty():

print("Cтек пуст")

else:

k = self.stack[-1]

self.stack.pop(len(self.stack)-1)

return k

def size(self):

return len(self.stack)

def peek(self):

return self.stack[-1]

def isFull(self):

if len(self.stack)==self.max\_size:

return 1

else:

return 0

s = Steck(VARIANT\*5 + 50)

while not s.isFull():

s.push(random.randint(1,1000+1))

s\_p = Steck()

s\_np = Steck()

while not s.empty():

k = s.pop()

if k % 2 == 0:

s\_p.push(k)

else:

s\_np.push(k)

print("Все числа из парного стека = ",end = "")

while not s\_p.empty():

print(s\_p.pop(),end = " ")

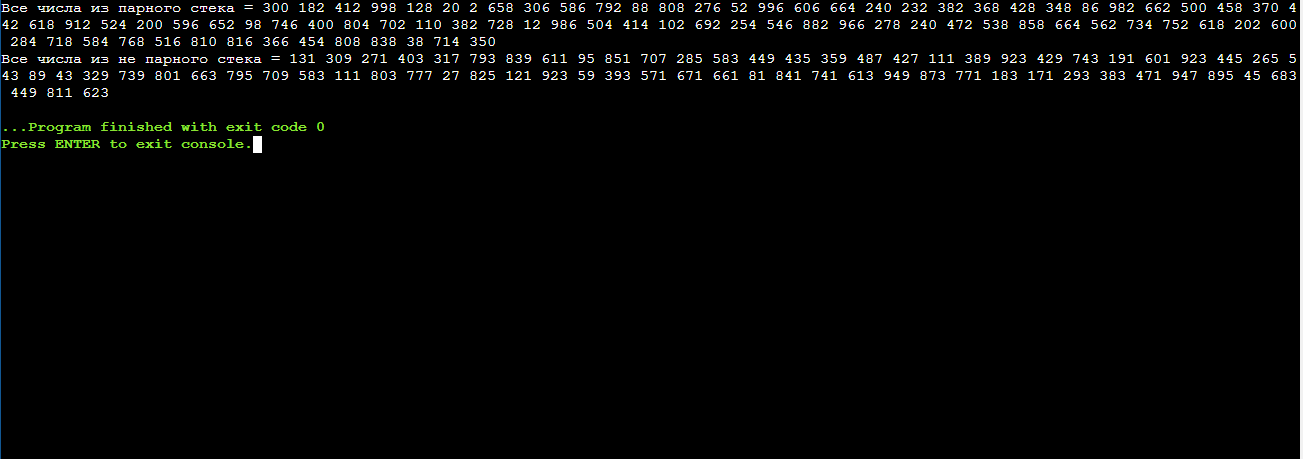
print()

print("Все числа из не парного стека = ",end = "")

while not s\_np.empty():

print(s\_np.pop(),end = " ")

Моя консоль



Мовою С++

#include <iostream>

#include <time.h>

using namespace std;

#define VARIANT 20

class Stack

{

int \*arr;

int t;

int capacity;

public:

Stack(int size){

arr = new int[size];

capacity = size;

t = -1;

}

~Stack(){

delete[] arr;

}

void push(int a){

t++;

arr[t] = a;

}

int pop(){

if(isEmpty()){

return t;

}

else{

return arr[t--];

}

}

int peek(){

if(isEmpty()){

return t;

}

else{

return arr[t];

return -1;

}

}

int size(){

return t+1;

}

bool isEmpty(){

return t == -1;

}

bool isFull(){

return t == capacity-1;

}

};

int main()

{

srand (time (NULL));

Stack s(VARIANT\*5+50);

while(!s.isFull()){

s.push(rand ()% 1000 + 1);

}

Stack s\_p(VARIANT\*5+50);

Stack s\_np(VARIANT\*5+50);

while(!s.isEmpty()){

int num = s.pop();

if(num%2==0){

s\_p.push(num);

}

else{

s\_np.push(num);

}

}

cout<<"Все парные числа = ";

while(!s\_p.isEmpty()){

cout<<s\_p.pop()<<" ";

}

cout<<endl;

cout<<"Все не парные числа = ";

while(!s\_np.isEmpty()){

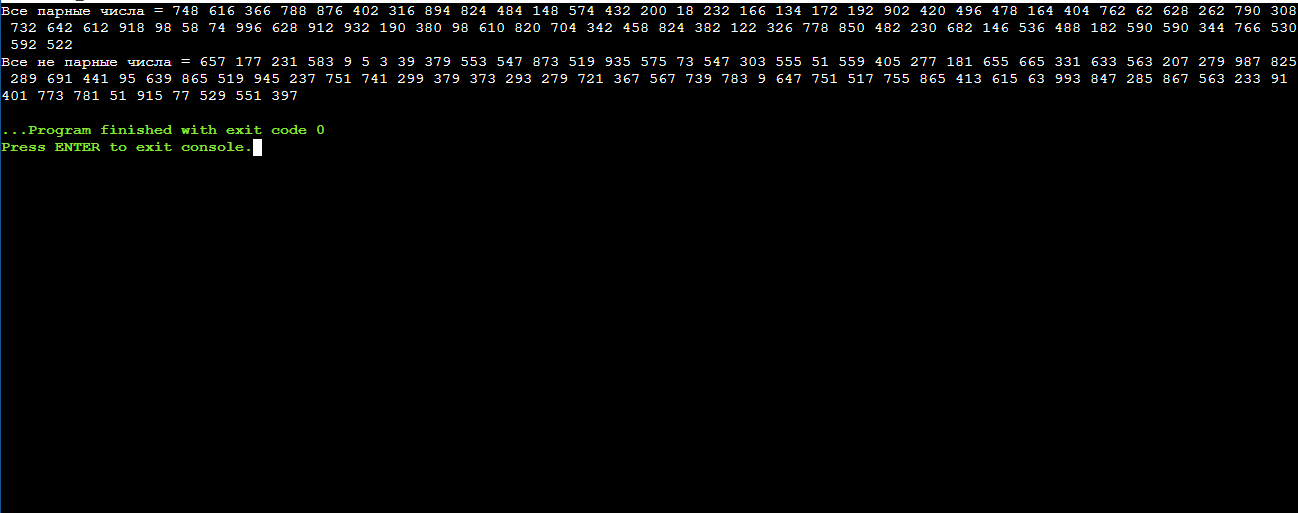
cout<<s\_np.pop()<<" ";

}

return 0;

}

Моя консоль



**Висновок:** У висновку я хочу сказати що виконуючи цю лабораторну роботи я остаточно зрозумів що таке стек, як користуватися стеком, та які у нього є методи. Також я самостійно навчився робити свій тип даних(Class) Steck, який повністю імітує реальний тип даних стек.