Assignment 2

-Problem 1-

Student: Wojcicki Andrei-Cristian

Grupa: 1.3B

Specializare: CR

Enuntul Problemei:

N copii se joaca jocul „ala-bala-portocala”. Ei sunt asezati in cerc si incep sa numere de la 1 la k. Al k-lea copil este scos din joc. Jocul continua cu restul copiiilor, de fiecare data numarand de la 1 la k, pana cand toti copiii sunt scosi din joc. Implementati un algoritm eficient prin care sa afisati ordinea in care copiii sunt scosi din joc folosind o lista circulara.

Pseudocodul algoritmului:

PUSH\_ELEMENT\_END(Lista, valoare\_noua)

Iterator = head

While iterator.next != NIL

Iterator = iterator.next

Last\_element = iterator

Last\_element.next = new\_element

New\_element = valoare\_noua

New\_element.next = NIL

ONE\_ELEMENT(Lista)

Iterator = head

If iterator.next == iterator

Returneaza 1

GENERATE\_NUMBERS()

N = random % 100000

Returneaza n

MAIN ( )

Head.next = NIL

n = generate\_numbers ()

for i = 1, n do

push\_element\_end (head,i)

while iterator.next != NIL

iterator = iterator.next

last\_element = iterator

last\_element.next = head.next

iterator = head

if k==1

for i = 1, n do

afiseaza valoarea iterator.next

iterator = iterator.next

else

while 1

if one\_element (iterator) == 1

break

for i = 1, k-1 do

iterator = iterator.next

afiseaza valoarea iterator.next

iterator.next = iterator.next.next

Descrierea programului:

Programul genereaza automat un numar random de copii. Apoi se creeaza prima oara o lista simpla inlantuita (adaugandu-se unul cate unul la final). Ca lista sa devina una circulara, ultimului nod i se taie legatura cu NULL-ul si se face una noua cu primul nod

( ).

Programul trateaza cazul particular in care k este 1. In acest caz, vor iesi pe rand fiecare copil, unul dupa altul. Daca k este diferit de 1, atunci copilul de pe pozitia k este afisat intr-un fisier si este scos din joc. Ascest lucru se va repeta atata timp cat mai exista copii in joc.

De exemplu daca sunt 10 copii, iar k este 2. Va rezulta la inceput:

1 2 3 4 5 6 7 8 9 10 -> se afiseaza si iese 2

1 3 4 5 6 7 8 9 10 -> se afiseaza si iese 4

1 3 5 6 7 8 9 10 -> se afiseaza si iese 6

1 3 5 7 8 9 10 -> se afiseaza si iese 8

1 3 5 7 9 10 -> se afiseaza si iese 10

1 3 5 7 9 -> se afiseaza si iese 3

1 5 7 9 -> se afiseaza si iese 7

1 5 9 -> se afiseaza si iese 1

5 9 -> se afiseaza si iese 9

5 -> se afiseaza si iese 5

La final se afiseaza 2 4 6 8 10 3 7 1 9 5

Pentru calcularea timpului de executie, se iau 2 variabile start la inceputul programului si finish la finalul acestuia. Timpul va fi retinut intr-o variabila duration care va fi diferenta dintre finish si start.

Pentru fiecare nod in parte se aloca o memorie de marimea unei structuri cu 2 campuri.

Programul genereaza un numar random de copii din intervalul [1, 1000000]