

Tecnológico Nacional de México  
Campus Querétaro.

# **Implemente Estructura de Colas (FIFO) en la simulación de atención de procesos.**

*Alumno:*

*Arellano Ochoa Daniel Ignacio*

*Carrera:*

*Ingeniera en sistemas computacionales*

*Docente:*

*Felipe Estrada Rojas*

*Fecha: 24/10/2020*

**Código:** El archivo cpp se anexo junto con este trabajo, de igual manera el código se encuentra hasta el final de este trabajo.

## Ejecución:

```
PS C:\Users\Daniel\Desktop\cola> .\principal
Bienvenido al programa, se ejecutaran 100 minutos de procesamiento simulado
Se generaron: 206 procesos
Se atendieron al 100%: 106 procesos
En la cola hay: 100 procesos
Lista de procesos incompletos:
Proceso 1 Porcentaje: 22
Proceso 2 Porcentaje: 14
Proceso 3 Porcentaje: 95
Proceso 4 Porcentaje: 93
Proceso 5 Porcentaje: 93
Proceso 6 Porcentaje: 8
Proceso 7 Porcentaje: 8
Proceso 8 Porcentaje: 87
Proceso 9 Porcentaje: 85
Proceso 10 Porcentaje: 32
Proceso 11 Porcentaje: 22
Proceso 12 Porcentaje: 99
Proceso 13 Porcentaje: 16
Proceso 14 Porcentaje: 16
Proceso 15 Porcentaje: 8
Proceso 16 Porcentaje: 8
Proceso 17 Porcentaje: 87
Proceso 18 Porcentaje: 85
Proceso 19 Porcentaje: 28
Proceso 20 Porcentaje: 22
Proceso 21 Porcentaje: 93
Proceso 22 Porcentaje: 8
Proceso 23 Porcentaje: 87
Proceso 24 Porcentaje: 85
Proceso 25 Porcentaje: 6
Proceso 26 Porcentaje: 16
Proceso 27 Porcentaje: 16
Proceso 28 Porcentaje: 24
Proceso 29 Porcentaje: 24
Proceso 30 Porcentaje: 87
Proceso 31 Porcentaje: 87
Proceso 32 Porcentaje: 6
Proceso 33 Porcentaje: 8
Proceso 34 Porcentaje: 16
Proceso 35 Porcentaje: 95
Proceso 36 Porcentaje: 93
Proceso 37 Porcentaje: 30
Proceso 38 Porcentaje: 85
Proceso 39 Porcentaje: 6
Proceso 40 Porcentaje: 8
Proceso 41 Porcentaje: 8
Proceso 42 Porcentaje: 93
Proceso 43 Porcentaje: 95
Proceso 44 Porcentaje: 93
Proceso 45 Porcentaje: 93
Proceso 46 Porcentaje: 22
Proceso 47 Porcentaje: 8
Proceso 48 Porcentaje: 8
Proceso 49 Porcentaje: 87
Proceso 50 Porcentaje: 85
Proceso 51 Porcentaje: 93
Proceso 52 Porcentaje: 14
Proceso 53 Porcentaje: 95
Proceso 54 Porcentaje: 93
Proceso 55 Porcentaje: 6
Proceso 56 Porcentaje: 8
Proceso 57 Porcentaje: 8
Proceso 58 Porcentaje: 87
Proceso 59 Porcentaje: 99
Proceso 60 Porcentaje: 24
Proceso 61 Porcentaje: 14
Proceso 62 Porcentaje: 95
Proceso 63 Porcentaje: 93
Proceso 64 Porcentaje: 93
Proceso 65 Porcentaje: 8
Proceso 66 Porcentaje: 8
Proceso 67 Porcentaje: 87
Proceso 68 Porcentaje: 85
Procesos con 0%: 32
PS C:\Users\Daniel\Desktop\cola>
```

En la ejecución anterior se generaron 206 procesos de los cuales 106 fueron completados al 100%, y de los 100 restantes 32 no fueron atendidos y 68 fueron atendidos por menos del 100%

```
Bienvenido al programa, se ejecutaran 100 minutos de procesamiento simulado
Se generaron: 301 procesos
Se atendieron al 100%: 88 procesos
En la cola hay: 213 procesos
Lista de procesos incompletos:
Proceso 1 Porcentaje: 97
Proceso 2 Porcentaje: 93
Proceso 3 Porcentaje: 80
Proceso 4 Porcentaje: 17
Proceso 5 Porcentaje: 80
Proceso 6 Porcentaje: 17
Proceso 7 Porcentaje: 93
Proceso 8 Porcentaje: 34
Proceso 9 Porcentaje: 80
Proceso 10 Porcentaje: 17
Proceso 11 Porcentaje: 93
Proceso 12 Porcentaje: 97
Proceso 13 Porcentaje: 93
Proceso 14 Porcentaje: 80
Proceso 15 Porcentaje: 17
Proceso 16 Porcentaje: 17
Proceso 17 Porcentaje: 93
Proceso 18 Porcentaje: 80
Proceso 19 Porcentaje: 34
Proceso 20 Porcentaje: 80
Proceso 21 Porcentaje: 17
Proceso 22 Porcentaje: 93
Proceso 23 Porcentaje: 93
Proceso 24 Porcentaje: 80
Proceso 25 Porcentaje: 17
Proceso 26 Porcentaje: 97
Proceso 27 Porcentaje: 17
Proceso 28 Porcentaje: 93
Proceso 29 Porcentaje: 80
Proceso 30 Porcentaje: 17
Proceso 31 Porcentaje: 93
Proceso 32 Porcentaje: 80
Proceso 33 Porcentaje: 97
Proceso 34 Porcentaje: 93
Proceso 35 Porcentaje: 80
Proceso 36 Porcentaje: 17
Proceso 37 Porcentaje: 17
Proceso 38 Porcentaje: 93
Proceso 39 Porcentaje: 80
Proceso 40 Porcentaje: 97
Proceso 41 Porcentaje: 80
Proceso 42 Porcentaje: 17
Proceso 43 Porcentaje: 93
Proceso 44 Porcentaje: 34
Proceso 45 Porcentaje: 80
Proceso 46 Porcentaje: 17
Proceso 47 Porcentaje: 93
Proceso 48 Porcentaje: 17
Proceso 49 Porcentaje: 93
Proceso 50 Porcentaje: 80
Proceso 51 Porcentaje: 97
Proceso 52 Porcentaje: 17
Proceso 53 Porcentaje: 93
Proceso 54 Porcentaje: 80
Proceso 55 Porcentaje: 34
Proceso 56 Porcentaje: 80
Proceso 57 Porcentaje: 17
Proceso 58 Porcentaje: 93
Proceso 59 Porcentaje: 93
Proceso 60 Porcentaje: 80
Proceso 61 Porcentaje: 17
Proceso 62 Porcentaje: 97
Proceso 63 Porcentaje: 93
Proceso 64 Porcentaje: 80
Proceso 65 Porcentaje: 17
Proceso 66 Porcentaje: 80
Proceso 67 Porcentaje: 17
Proceso 68 Porcentaje: 93
Proceso 69 Porcentaje: 34
Proceso 70 Porcentaje: 93
Proceso 71 Porcentaje: 80
Proceso 72 Porcentaje: 17
Proceso 73 Porcentaje: 97
Proceso 74 Porcentaje: 93
Proceso 75 Porcentaje: 80
Proceso 76 Porcentaje: 17
Proceso 77 Porcentaje: 17
Proceso 78 Porcentaje: 93
Proceso 79 Porcentaje: 80
Proceso 80 Porcentaje: 97
Proceso 81 Porcentaje: 17
Proceso 82 Porcentaje: 93
Proceso 83 Porcentaje: 80
Proceso 84 Porcentaje: 93
Proceso 85 Porcentaje: 80
Proceso 86 Porcentaje: 17
Proceso 87 Porcentaje: 97
Proceso 88 Porcentaje: 17
Proceso 89 Porcentaje: 93
Procesos con 0%: 124
PS C:\Users\Daniel\Desktop\cola>
```

En la ejecución anterior se generaron 301 procesos de los cuales 88 fueron completados al 100%, y de los 213 restantes, 124 no fueron atendidos y 89 fueron atendidos por menos del 100%

## Conclusión:

El uso de las pilas puede llegar a ser muy útil como herramienta al momento de realizar algunos programas específicos que aunque pueden ser solucionados ocupando otras herramientas como pilas o listas, es importante saber en qué momentos se ocupan las colas pues este tipo de estructuras manejan ciertos tipos de funciones que la hacen óptima para solucionar algunos problemas.

## Código:

```
/*Nombre: Arellano Ochoa Daniel Ignacio
No. de control: 19141118
Programa que simula el procesamiento durante 100 minutos en los cuales por c
ada minutos se genraran n procesos y se atenderan m procesos donde m y n son
valores aletarios de 0 a 5
*/
#include <string>
#include <iostream>
#include "Procesador.hpp"
#include "Proceso.hpp"

using namespace std;

//Menu de inicio del programa
int main(int argc, const char * argv[]){

    cout<<"Bienvenido al programa, se ejecutaran 100 minutos de procesamient
o simulado"<<endl;
    Procesador* intel=new Procesador();
    intel->procesar();

}
```

```
#ifndef Proceso_hpp
#define Proceso_hpp

using namespace std;
class Proceso
{
private:
    int porcentaje;
    Proceso* ant;
```

```

public:
    Proceso();
    int getPorcentaje();
    Proceso* getAnt();
    void setPorcentaje(int*);
    void setAnt(Proceso*);

};

#endif /*Proceso_hpp*/

```

```

#include "Proceso.hpp"
#include <iostream>
#include <string>
//clase de proceso
using namespace std;
//constructor con los atributos de porcentaje y ant inicializados
Proceso::Proceso(){
    this->porcentaje=0;//porcentaje atendido
    this->ant=NULL;//procesos atras de este dentro de una cola
}
//metodo que regresa el porcentaje que ha sido atendido este proceso
int Proceso::getPorcentaje(){
    return this->porcentaje;
}
//metodo que refresa el proceso que se encuentra anterior a este dentro de una cola
Proceso* Proceso::getAnt(){
    return this->ant;
}
//metodo que permite modificar el proceso que se encuentra atras de este dentro de una cola
void Proceso::setAnt(Proceso* p){
    this->ant=p;
}
//metodo que permite modificar el porcentaje atendido de este proceso
void Proceso::setPorcentaje(int* n){
    this->porcentaje=*n;
}

```

```

#ifndef Procesador_hpp
#define Procesador_hpp
#include "Proceso.hpp"

```

```

class Procesador
{
private:
    Proceso* inicio;
    Proceso* final;
    int proGenerados;
    int proAtendidos;
    int proCola;
    int procero;

public:
    Procesador();

    void insertar(Proceso*);
    void sacar();
    void generarProcesos(int*);
    void atenderProcesos(int*);
    void procesar();
    void procesosColaInc();

};

#endif /*Procesador_hpp*/

```

```

#include "Procesador.hpp"
#include "Proceso.hpp"
#include <string>
#include <iostream>
#include <stdlib.h>
#include <time.h>
//clase de procesador
using namespace std;
//constructor del procesador, con datos inicializados
Procesador::Procesador(){
    this->inicio=NULL;//proceso que se encuentra al inicio de la cola
    this->final=NULL;//procesos que se encuetnra al final de la cola
    this->proGenerados=0;//procesos generados
    this->proAtendidos=0;//procesos atendido al 100%
    this->proCola=0;//procesos atendidos de 0% a 99%
    this->procero=0;//procesos no atendido
}

```

```

//metodo para insertar un procesos dentro de la cola
void Procesador::insertar(Proceso* p){
    if(this->proCola==0){
        this->inicio=p;
        this->final=p;
    }else{
        this->final->setAnt(p);
        this->final=p;
    }
    this->proCola++;
}

//metodo para sacar el proceso que se encuentra al inicio de la cola
void Procesador::sacar(){
    this->inicio=this->inicio->getAnt();
    this->proCola--;
}

//metodo que generar N procesos, insertandolos a la cola
void Procesador::generarProcesos(int* N){
    Proceso* pro;
    for (size_t i = 0; i < *N; i++)
    {
        pro=new Proceso();
        this->insertar(pro);
        this->proGenerados++;
    }
}

//metodo que atiende M procesos, verificando si se logro atender al 100%, si
//es el caso contrario se volvera a formar dentro de la cola
void Procesador::atenderProcesos(int* M){
    srand(time(NULL));
    int pAux;
    Proceso* proAux;
    for (size_t i = 0; i < *M; i++)
    {
        if(this->proCola>0){
            proAux=this->inicio;
            this->sacar();
            pAux=rand()%100 + 1;
            pAux= pAux + proAux->getPorcentaje();
            proAux->setPorcentaje(&pAux);

            if(proAux->getPorcentaje()<100){
                this->insertar(proAux);
            }else{

```

```

        this->proAtendidos++;
    }

}

}

}

//metodo que generar y atiende N y M procesos respectivamente, durante 100
ciclos (100 minutos simulados), dando informacion al final
void Procesador::procesar(){
    srand(time(NULL));
    int M;//atender
    int N;//generar

    for (size_t i = 0; i < 100; i++)
    {
        //cout<<"minuto"<<i<<endl;
        M= rand()%6;
        N= rand()%6;
        //cout<<"M= "<<M<<" N= "<<N<<endl;
        this->generarProcesos(&N);
        if(M>this->proCola){
            M=this->proCola;
            this->atenderProcesos(&M);
        }else{
            this->atenderProcesos(&M);
        }
    }

    cout<<"Se generaron: "<<this->proGenerados<<" procesos"<<endl;
    cout<<"Se atendieron al 100%: "<<this->proAtendidos<<" procesos"<<endl;
    cout<<"En la cola hay: "<<this->proCola<<" procesos"<<endl;
    cout<<"Lista de procesos incompletos:"<<endl;
    this->procesosColaInc();
}

//metodo que cuenta los procesos que quedaron sin atender y enumero e imprim
e los procesos que fueron atendidos pero no al 100%
void Procesador::procesosColaInc(){
    Proceso* auxPro;
    auxPro=this->inicio;
    int i=1;
    bool auxF=true;
    while (auxPro!=NULL && auxF==true)

```

```
{
    if(auxPro == this->final)
        auxF=false;
    if(auxPro->getPorcentaje()>0){
        cout<<"Proceso "<<i<<" Porcentaje: "<<auxPro-
>getPorcentaje()<<endl;
        i++;
    }else{
        this->procero++;
    }
    auxPro=auxPro->getAnt();
}
cout<<"Procesos con 0%: "<<this->procero<<endl;
}
```