```
/*CSC 326 Lab 4 Queue*/
#include<iostream>
#include<fstream>
#include<vector>
#include<string>
#include<queue>

using namespace std;

class Queue {
private:
	int head = 0;
	int tail = 0;
	int capacity = 0;
	int count = 0;
	int* arr;

public:
	Queue(int capacity);
	void Enqueue(char process, int srvct, int trnrndt);
	int Dequeue();
	~Queue();
};
Queue::~Queue() {
	delete[] arr;
}
Queue::Queue(int capacity) : capacity(capacity) {
	arr = new int[capacity];
	//this->capacity = capacity;
}
void Queue::Enqueue(char process, int srvct, int trnrndt) {
	if (count == capacity) {
		cout << "Queue is already full" << endl;
		return;
	}
}
int Queue::Dequeue() {
	if (count == 0) {
		cout << " Queue is currently empty." << endl;
		return 0;
	}
	int temp = arr[head];
	head = (head + 1) % capacity;
	count--;
	return temp;
}
int main()
{
	Queue process(5);
	Queue service_time(5);
	Queue turnaround_time(5);

	process.Enqueue(A, 0, 3);
	process.Enqueue(B, 2, 6);
	process.Enqueue(C, 4, 4);
	process.Enqueue(D, 6, 5);
	process.Enqueue(E, 8, 2);
```

```
	/*Queue process(5);
	Queue service_time(5);
	Queue turnaround_time(5);
	ifstream fin;
	fin.open("jobs.txt");
		if (!fin.fail()){
			return 1;
		}
	char job;
	int srvct, trnrndt;
	while (!fin.eof()) {
		fin >> job >> srvct >> trnrndt;
		process.Enqueue(job);
		service_time.Enqueue(srvct);
		turnaround_time.Enqueue(trnrndt);
		cout << "Process: " << job << endl;
		cout << "service time: " << srvct << endl;
	}
	*/
	system("pause>0");
}
```