Practical 14

#include <iostream>

using namespace std;

#define size 5

class pizza

{

int porder[size];

int front, rear;

public:

pizza()

{

front = rear =1;

}

int qfull()

{

if ((front == 0) && (rear == (size - 1)) || (front == (rear + 1) % size))

return 1;

else return 0;

}

int qempty()

{

if (front == -1)

return 1;

else return 0;

}

void accept\_order(int);

void make\_payment(int); void order\_in\_queue();

};

void pizza::accept\_order(int item)

{

if (qfull()) cout << "\nVery Sorry !!!! No more orders....\n";

else

{

if (front == -1)

{ front=rear = 0;

}

else

{

rear = (rear + 1) % size;

} porder[rear] = item;

}

} void pizza::make\_payment(int n)

{

int item;

char ans;

if (qempty())

cout << "\nSorry !!! order is not there...\n";

else

{ cout << "\n Delivered orders as follows...\n"; for (int i = 0; i < n; i++)

{

item = porder[front]; if (front == rear)

{

front = rear = -1;

}

else

{

front = (front + 1) % size;

} cout << "\t" << item;

}

cout << "\nTotal amount to pay : " << n \* 100; cout << "\nThank you visit Again....\n";

}

} void pizza::order\_in\_queue()

{

int temp;

if (qempty())

{ cout << "\nSorry !! There is no pending order...\n";

}

else

{

temp = front; cout << "\nPending Order as follows..\n";

while (temp != rear)

{

cout << "\t" << porder[temp]; temp = (temp + 1) % size;

} cout << "\t" << porder[temp];

}

}

int main()

{

pizza p1;

int ch, k, n;

do

{

cout << "\n\t\*\*\*\*\* Welcome To Pizza Parlor \*\*\*\*\*\*\*\n";

cout << "\n1.Accept order\n2.Make\_payment\n3.Pending Orders\nEnter u r choice: ";

cin >> ch;

switch (ch)

{ case 1:

cout << "\nWhich Pizza do u like most....\n";

cout << "\n1.Veg Soya Pizza \n2.Veg butter Pizza\n3.Egg\_Pizza"; cout << "\nPlease enter u r order: ";

cin >> k;

p1.accept\_order(k);

break;

case 2:

cout << "\nHow many Pizza ?";

cin >> n; p1.make\_payment(n);

break;

case 3: cout << "\n Following orders are in queue to deliver....as follows..\n"; p1.order\_in\_queue();

break;

}

}

while (ch != 4); return 0;

}

