Screen shot with all the warnings

xxxx

Screen shot with first warning fixed: line 95 \* headPtr = newPtr

Program has crashed:

Xxx

Output when the 2nd warning

Line 101 \*tailPtr = newPtr

Xxx

Screen shot when all the warnings are fixed

113 \*headPtr = (\*headPtr) ->nextPtr in the dequeue function

Xxx

Explaining:

Warning 1: headPtr = newPtr;

The first mistake is this headPtr is local to the function this headPtr is assigned to the address of the new node, but it should been heatPtr in the main pointing to the new node.

Also we know the headPtr in the main contains null so that’s why the list is empty, to fix this problem I just changed headPtr to \*headPtr

\*headPtr in main must be assigned to point to the new node by doing this now we can add an element to the queue.

Fixed : \*headPtr = newPtr;

Warning 2: tailPtr = newPtr

So by fixing the first warning we create and added the first node which \*headPtr in the main pointing to it but the tailPtr should also point to it too, but it only point to the local tailPtr, which this tailPtr in the main and it is still null, so when we add the second node program crashes as tailPtr has a null value so problem will fix by changing tailPtr to \*tailPtr.

Fixed : \*tailPtr = newPtr;

Warning 3: headPtr = (\*headPtr) ->nextPtr

When we call the dequeue function the RHS will assign the address of the second node to the headPtr, and the free freePtr roves the first node so\*headPtr, so when we call the dequeue function the program crashes. By changing the headPtr to the \*headPtr this warning would be fixed so the headPtr will point to the second node in the queue and we can remove a node from the queue.

Fixed : \*headPtr = (\*headPtr) -> nextPtr;