
Práctico 2: Agregando condiciones compuestas y Filas

Objetivos:

Al finalizar este práctico se espera que los alumnos:

- comprendan la estructura “fila” y sus operaciones asociadas,
- reconozcan las principales diferencias que existen entre las pilas y las filas,
- puedan decidir, dependiendo del problema, qué estructuras utilizar para resolverlo.
- puedan aplicar las condiciones compuestas

Codificar los siguientes programas leyendo desde teclado o inicializando las pilas según corresponda. En caso de ser necesario se pueden utilizar estructuras auxiliares.

- 1) Pasar los elementos de la fila ORIGEN leída por teclado a la pila DESTINO inicialmente vacía.
- 2) Invertir el orden de la fila DADA leída por teclado.
- 3) Cargar por teclado la fila DATOS y pasar el primer elemento de la DATOS a su última posición, dejando los restantes elementos en el mismo orden. Finalmente muestre por pantalla.
- 4) Cargar por teclado la fila DATOS y pasar el último elemento de DATOS a su primera posición dejando los restantes elementos en el mismo orden. Finalmente muestre la fila por pantalla.
- 5) Dada una fila no vacía ORIGEN, pasar todos los elementos DESTINO hasta que aparezca un cero u ORIGEN quede sin elementos.
- 6) Realizar un programa para comparar la cantidad de elementos de las filas A y B. Si son iguales dejar el tope de la pila Resultado en la pila VERDADERO y son distintas en la pila FALSO.
- 7) Comparar las filas A y B para determinar si son iguales, es decir que tienen la misma cantidad de elementos y en el mismo orden. Si son iguales, dejar el tope de la pila RESULTADO en la pila IGUALES y si son distintas en DISTINTAS.
- 8) La fila DADA se encuentra ordenada de menor a mayor. Insertar el tope de la pila ELEMENTO en DADA de tal forma que continúe ordenada.
- 9) Dada una fila ORIGEN que se lee por teclado y una fila MODELO inicializada con un sólo elemento, modificar ORIGEN tal que si existe un elemento igual al de MODELO se lo ubique al principio de la estructura, considerando que sólo cambia la primer ocurrencia de ese elemento. Imprima ORIGEN.
- 10) Ubicar el primer elemento de la fila DADA detrás del primer elemento de valor 12, dejando los demás elementos en el mismo orden. Si el elemento de valor 12 es el primero, dejarlo como estaba. Se supone que al menos hay un elemento de valor de 12.
- 11) Concatenar dos pilas de modo que la que posee menos elementos quede abajo; si ambas tienen la misma cantidad de elementos, cualquiera puede quedar abajo.
- 12) Indicar si una pila PARTE está incluida en otra pila GRANDE (o sea, que una porción de GRANDE es igual a PARTE con los mismos elementos en el mismo orden). La forma de indicarlo será dejando vacía PARTE si efectivamente está incluida.

13) Para este problema: “Dada una fila ATENCIONCLIENTES que se lee por teclado imprimir por pantalla si está ordenada o no de menor a mayor. ATENCIONCLIENTES debe quedar con todos los elementos y en el mismo orden.”, se han escrito estas dos soluciones:

Solución A

```

1 Program ATENCIONClientesEstaOrdenado;
2 {Dada una fila ATENCIONCLIENTES que se lee por teclado mostrar por pantalla si está ordenada
3 ó no de menor a mayor. ATENCIONCLIENTES debe quedar con todos los elementos y en el mismo orden.
4 {$INCLUDE/IntroProg/Estructu}
5 var
6     ATENCIONCLIENTES, aux, auxNO:Fila;
7 begin
8     ReadFila(ATENCIONCLIENTES);
9     InicFila(aux, '');
10    InicFila(auxNO, '');
11    agregar(aux, extraer(ATENCIONCLIENTES));
12    while not FilaVacía(ATENCIONCLIENTES) do begin
13        if Primero(aux) < Primero(ATENCIONCLIENTES) then
14            agregar(aux, extraer(ATENCIONCLIENTES))
15        else
16            agregar(auxNO, extraer(ATENCIONCLIENTES));
17    end;
18    if not FilaVacía(auxNO) then
19        write('La fila no esta ordenada de menor a mayor.')
20    else
21        write('La fila está ordenada de menor a mayor. ');
22 end.

```

Activar Windows

Solución B

```

1 program ejercicio13;
2 {$INCLUDE/IntroProg/Estructu}
3 var ATENCIONCLIENTES, SI, NO, Resultado:Fila;
4 var aux, aux2:Pila;
5 begin
6     readfila(ATENCIONCLIENTES);
7     inicpila(aux, '');
8     inicpila(aux2, '');
9     inicfila(SI, '');
10    inicfila(NO, '');
11    inicfila(Resultado, '0');
12    if not filavacia(ATENCIONCLIENTES) then
13        begin
14            apilar(aux, extraer(ATENCIONCLIENTES));
15            while not filavacia(ATENCIONCLIENTES) and not (tope(aux) > primero(ATENCIONCLIENTES)) do
16                apilar(aux, extraer(ATENCIONCLIENTES));
17        end;
18    if filavacia(ATENCIONCLIENTES) then
19        agregar(SI, extraer(Resultado))
20    else
21        agregar(NO, extraer(Resultado));
22    writefila(SI);
23    writefila(NO);
24    while not pilavacia(aux) do
25        apilar(aux2, desapilar(aux));
26    while not pilavacia(aux2) do
27        agregar(ATENCIONCLIENTES, desapilar(aux2));
28    writefila(ATENCIONCLIENTES);
29 end.

```

Responder para ambas Soluciones

- a) ¿El programa cumple su objetivo? Si, no ¿por qué?
- b) Si no lo cumple, modificalo para que lo haga.
- c) ¿Existe alguna otra mejora que le podés hacer?

14) Modificar el programa del ejercicio 9 para reubicar todas las ocurrencias del elemento de MODELO que aparecen en ORIGEN.