

Universidad de las Fuerzas Armadas ESPE

Nombre: Luis Sagnay

NRC: 14543

Fecha: 16-06-2024

Funcionamiento e Implementación de Pilas

Descripción

Implementar una aplicación que utilice pilas para clasificar números ingresados por el usuario. La aplicación deberá permitir al usuario ingresar varios números a una pila para luego, utilizando pilas, clasificar los números primos en una pila, los números pares en otra pila y los números impares no primos en otra pila. La impresión final deberá constar de la pila original, la pila de números primos, la pila de números pares y la pila de números impares.

Implementación

Definición de Pilas y Variables

En el programa, se definen cuatro pilas utilizando arreglos de tamaño fijo ($MAX = 20$):

- **original:** Almacena los números ingresados por el usuario.
- **primos:** Almacena los números identificados como primos.
- **pares:** Almacena los números identificados como pares.
- **impares:** Almacena los números identificados como impares no primos.

Cada pila tiene asociada una variable cima que indica el índice del último elemento agregado a la pila.

Funciones

- **estaVacia:** Función booleana que verifica si una pila está vacía. Devuelve true si cima es igual a -1, indicando que la pila no tiene elementos.
- **estaLlena:** Función booleana que verifica si una pila está llena. Devuelve true si cima es igual a $MAX - 1$, indicando que la pila ha alcanzado su capacidad máxima.
- **insertar:** Inserta un valor dado en la pila especificada por pila. Incrementa cima antes de agregar el valor para indicar que se ha agregado un nuevo elemento.
- **quitar:** Quita y devuelve el último valor agregado a la pila especificada por pila. Decrementa cima después de devolver el valor para indicar que se ha eliminado un elemento.
- **esPrimo:** Verifica si un número dado es primo utilizando métodos convencionales de prueba de primalidad. Retorna true si el número es primo y false si no lo es.

Clasificación

Ingreso de Números

El programa solicita al usuario que ingrese números uno por uno hasta que ingrese -1 para finalizar, donde cada número ingresado se inserta en la pila original y en la pila copia para posteriormente ser impresa.

Se implementa una función **clasificarNumeros** que clasifica los números de la pila original en las pilas primos, pares e impares.

Para cada número sacado de la pila original:

Se verifica si es primo utilizando la función **esPrimo**.

Si es primo, se inserta en la pila primos.

Si no es primo, pero es par, se inserta en la pila pares.

Si es impar y no primo, se inserta en la pila impares.

Impresión de Resultados

Después de clasificar los números, se imprimen los contenidos de cada una de las pilas:

- **original:** Contiene los números ingresados por el usuario en el orden en que fueron ingresados.
- **primos:** Contiene los números identificados como primos.
- **pares:** Contiene los números identificados como pares.
- **impares:** Contiene los números identificados como impares que no son primos.

Conclusión

Este programa demuestra una aplicación práctica de las pilas para clasificar y organizar datos de manera eficiente. La implementación permite una fácil manipulación y gestión de los números ingresados por el usuario, facilitando su clasificación en diferentes categorías según criterios específicos como el primalidad y la paridad.