Currency Converter in C

#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
  
// Currency data  
char \*currencies[] = {"USD", "EUR", "GBP"};  
float rates[] = {0.012, 0.011, 0.0095}; // 1 INR to USD, EUR, GBP  
int numCurrencies = 3;  
  
// Linked list to store history  
typedef struct History {  
 float inrAmount;  
 char currency[4];  
 float convertedAmount;  
 struct History\* next;  
} History;  
  
History\* head = NULL;  
  
// Function to add to history  
void addToHistory(float inr, char\* currency, float converted) {  
 History\* newEntry = (History\*)malloc(sizeof(History));  
 newEntry->inrAmount = inr;  
 strcpy(newEntry->currency, currency);  
 newEntry->convertedAmount = converted;  
 newEntry->next = head;  
 head = newEntry;  
}  
  
// Function to display conversion history  
void showHistory() {  
 History\* temp = head;  
 printf("\n--- Conversion History ---\n");  
 while (temp != NULL) {  
 printf("%.2f INR to %s = %.2f\n", temp->inrAmount, temp->currency, temp->convertedAmount);  
 temp = temp->next;  
 }  
}  
  
// Main conversion function  
void convertCurrency() {  
 float amount;  
 int choice;  
  
 printf("Enter amount in INR: ");  
 scanf("%f", &amount);  
  
 printf("Choose currency to convert to:\n");  
 for (int i = 0; i < numCurrencies; i++) {  
 printf("%d. %s\n", i + 1, currencies[i]);  
 }  
 scanf("%d", &choice);  
  
 if (choice < 1 || choice > numCurrencies) {  
 printf("Invalid choice.\n");  
 return;  
 }  
  
 float converted = amount \* rates[choice - 1];  
 printf("Converted Amount: %.2f %s\n", converted, currencies[choice - 1]);  
  
 addToHistory(amount, currencies[choice - 1], converted);  
}  
  
int main() {  
 int option;  
  
 do {  
 printf("\n--- Currency Converter ---\n");  
 printf("1. Convert Currency\n");  
 printf("2. Show History\n");  
 printf("3. Exit\n");  
 printf("Choose an option: ");  
 scanf("%d", &option);  
  
 switch(option) {  
 case 1:  
 convertCurrency();  
 break;  
 case 2:  
 showHistory();  
 break;  
 case 3:  
 printf("Exiting...\n");  
 break;  
 default:  
 printf("Invalid option.\n");  
 }  
 } while(option != 3);  
  
 return 0;  
}