Assignment-20 (Pointers)

1. Write a function to swap values of two in variables of calling function. (TSRS)

#include<stdio.h>

void swapNums(int \*num1, int \*num2)

{

\*num1 = \*num1 + \*num2;

\*num2 = \*num1 - \*num2;

\*num1 = \*num1 - \*num2;

}

int main()

{

int a = 34, b = 79;

printf("Numbers before swapping are:\na = %d and b = %d\n", a, b);

swapNums(&a, &b);

printf("Numbers after swapping are:\na = %d and b = %d\n", a, b);

return 0;

}

2. Write a function to swap strings of two char arrays of calling functions. (TSRS)

#include<stdio.h>

#include<string.h>

char \* maxLengthStringBetween2Strings(char s1[], char s2[])

{

int lens1 = strlen(s1), lens2 = strlen(s2);

if(lens1 > lens2)

return s1;

else

return s2;

}

void swapStrings(char \*s1, char \*s2)

{

char temp[strlen(maxLengthStringBetween2Strings(s1, s2))];

strcpy(temp, s1);

strcpy(s1, s2);

strcpy(s2, temp);

}

int main()

{

char str1[] = {"Hello"}, str2[] = {"World"};

printf("Strings before swapping are:\nstring 1: %s and string 2: %s\n", str1, str2);

swapStrings(str1, str2);

printf("Strings after swapping are:\nstring 1: %s and string 2: %s\n", str1, str2);

return 0;

}

3. Write a function to sort an array of int type values. [ void sort(int \*ptr,int size); ]

#include<stdio.h>

void swapNums(int \*num1, int \*num2)

{

\*num1 = \*num1 + \*num2;

\*num2 = \*num1 - \*num2;

\*num1 = \*num1 - \*num2;

}

void sort(int \*ptr, int size)

{

int i, count = -1;

while(count)

{

count = 0;

for(i = 0; i < size - 1; i++)

if(\*(ptr + i) > \*(ptr + i + 1))

{

swapNums(ptr + i, ptr + i + 1);

count++;

}

}

}

int main()

{

int arr[10], i;

printf("Enter 10 numbers: ");

for(i = 0; i < 10; i++)

scanf("%d", &arr[i]);

sort(arr, 10);

printf("Array after sorting is now:\n");

for(i = 0; i < 10; i++)

printf("%d ", arr[i]);

return 0;

}

4. Write a program in C to demonstrate how to handle the pointers in the program.

#include<stdio.h>

int main()

{

int x = 10, \*p, \*\*q, \*\*\*r;

p = &x;

q = &p;

r = &q;

printf("%d %d %d %d\n", x, \*p, \*\*q, \*\*\*r);

printf("%d %d %d %d\n", &x, p, \*q, \*\*r);

printf("%d %d %d\n", &p, q, \*r);

printf("%d %d\n", &q, r);

printf("%d\n", &r);

return 0;

}

5. Write a program to find the maximum number between two numbers using a pointer

#include<stdio.h>

int main()

{

double num1, num2, \*greatest;

printf("To check the greater number between two numbers,\nEnter first number: ");

scanf("%lf", &num1);

printf("Enter second number: ");

scanf("%lf", &num2);

if(num1 > num2)

greatest = &num1;

else

greatest = &num2;

printf("The maximum number between %lf and %lf is %lf.", num1, num2, \*greatest);

return 0;

}

6. Write a program to calculate the length of the string using a pointer

#include<stdio.h>

int main()

{

char str[50];

int i;

printf("Enter string: ");

gets(str);

for(i = 0; \*(str + i); i++);

printf("Length of %s is %d.", str, i);

return 0;

}

7. Write a program to count the number of vowels and consonants in a string using a pointer.

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int vowelsCountInString(char \*);

int vowelsCountInString(char \*str)

{

int vowels\_count = 0, i;

for(i = 0; \*(str + i); i++)

if(str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' || str[i] == 'u' || str[i] == 'A' || str[i] == 'E' || str[i] == 'I' || str[i] == 'O' || str[i] == 'U')

vowels\_count++;

return vowels\_count;

}

int main()

{

char str[50], ch;

int i, count = 0;

printf("Enter a string: ");

fgets(str, 49, stdin);

str[strlen(str) - 1] = '\0';

printf("Number of vowels in %s are %d.\n", str, vowelsCountInString(str));

printf("Number of consonants in %s are %d.\n", str, strlen(str) - vowelsCountInString(str));

return 0;

}

8. Write a program to compute the sum of all elements in an array using pointers.

#include<stdio.h>

int main()

{

int arr[10], i, sum = 0;

printf("Enter 10 numbers: ");

for(i = 0; i < 10; i++)

scanf("%d", &arr[i]);

for(i = 0; i < 10; i++)

sum = sum + (\*(arr + i));

printf("Sum of all 10 numbers of array is %d.", sum);

return 0;

}

9. Write a program to print the elements of an array in reverse order.

#include<stdio.h>

int main()

{

int n, arr[n], i;

printf("Enter number of elements to be stored in array: ");

scanf("%d", &n);

printf("Enter %d values to be stored in array: ", n);

for(i = 0; i < n; i++)

scanf("%d", &arr[i]);

printf("Array values in reverse order are:\n");

for(i = n - 1; i >= 0; i--)

printf("%d ", arr[i]);

return 0;

}

10. Write a program to print a string in reverse using a pointer

#include<stdio.h>

#include<string.h>

void swapNums(char \*num1, char \*num2)

{

\*num1 = \*num1 + \*num2;

\*num2 = \*num1 - \*num2;

\*num1 = \*num1 - \*num2;

}

int main()

{

char str[50];

int i, str\_length = 0, last\_num;

printf("Enter a string to reverse: ");

fgets(str, 50, stdin);

str[strlen(str) - 1] = '\0';

for(i = 0; \*(str + i); i++)

str\_length++;

last\_num = str\_length - 1;

for(i = 0; i < str\_length / 2; i++)

{

swapNums(str + i, str + last\_num);

last\_num--;

}

printf("String after reversing is %s", str);

return 0;

}