פתרונות תרגילים נוספים ברקורסיה:

תרגיל 1:

int DigitSum(int n1)

{

if(n1 == 0)

return 0;

return ((n1 % 10) + DigitSum(n1 / 10));//calling the function DigitSum itself

}

תרגיל 2:

char\* ReverseOfString(char str1[])

{

static int i=0;

static char revstr[MAX];

if(\*str1)

{

ReverseOfString(str1+1);//calling the function ReverseOfString itself

revstr[i++] = \*str1;

}

return revstr;

}

תרגיל 3:

long convertBinary(int decNo)

{

static long biNo,r,fctor = 1;

if(decNo != 0)

{

r = decNo % 2;

biNo = biNo + r \* fctor;

fctor = fctor \* 10;

convertBinary(decNo / 2);//calling the function convertBinary itself recursively

}

return biNo;

}

תרגיל 4:

int checkPrime(int n1,int i)

{

if(i==1)

{

return 1;

}

else

{

if(n1 %i==0)

return 0;

else

checkForPrime(n1,i-1);//calling the function checkForPrime itself recursively

}

}

תרגיל 5:

void checkPalindrome(char wordPal[], int index)

{

int len = strlen(wordPal) - (index + 1);

if (wordPal[index] == wordPal[len])

{

if (index + 1 == len || index == len)

{

printf(" The entered word is a palindrome.\n\n");

return;

}

checkPalindrome(wordPal, index + 1);//calling the function itself recursively

}

else

{

printf(" The entered word is not a palindrome.\n\n");

}

}

תרגיל 6:

void copyString(char stng1[], char stng2[], int ctr)

{

stng2[ctr] = stng1[ctr];

if (stng1[ctr] == '\0')

return;

copyString(stng1, stng2, ctr + 1);

}