PDS Lab
Assignment 4, Recursion

September 3, 2024

Total Marks: 100

Instructions

- Create a directory named as Lab-4.

- Give the name of the program as $< p.c >$ where $< p >$ implies the problem number, like 1.c, 2.c, 3.c, etc. Store all the programs of this week under this directory.

- You should upload all .c files (1.c, 2.c, 3.c) as a single Zip file to the Moodle course web page latest by 5.00 PM (without penalty). The cutoff time will be till 5.15 PM with a penalty of 25% on your secured marks (i.e., if you secured 80 marks, after penalty you will get 60 marks). Beyond 5.15 PM, the moodle system will not allow you to submit, as a result you will get zero.

---

1. The Recamán's sequence $a_0, a_1, a_2, \ldots$ is defined as: (35)

$$a_n = \begin{cases} 0 & \text{if } n = 0 \\ a_{n-1} - n & \text{if } a_{n-1} - n > 0 \text{ and not already present in the sequence.} \\ a_{n-1} + n & \text{otherwise} \end{cases}$$

Write a *recursive* function to print the first n terms in the sequence. You may use the following template to write your code.

```c
#include <stdio.h>

int recaman(int n, int arr[]) {
        // Complete this recursive function
}

int main(void) {
    int num;
    scanf("%d", &num);
    int arr[num];

    recaman(num, arr);
    for (int i = 0; i < num; i++) {
        printf("%d ", arr[i]);
    }
}
```

```
    }

    return 0;
}
```

For reference, the following are the first 75 terms in the sequence.

```
 1  3  6  2  7 13 20 12 21 11 22 10 23  9 24  8 25 43 62 42 63 41 18
42 17 43 16 44 15 45 14 46 79 113 78 114 77 39 78 38 79 37 80
36 81 35 82 34 83 33 84 32 85 31 86 30 87 29 88 28 89 27 90 26
91 157 224 156 225 155 226 154 227 153 228
```

2. In a galaxy far, far away a fierce battle is raging between the Resistance and the   (35)
   sinister Empire. As a member of the Resistance, you've intercepted the following
   message from the captain of a spacecraft just before it was captured by the Empire.

   ```
   .epoh ylno ym er'uoy .ibonek naw-ibo ,em pleh
   ```

   Your mission is to decrypt this message and uncover its true meaning in English.
   Write a *recursive* function that takes the above string as input and prints the de-
   crypted text to the console. You **must** not declare a new array for the decryption
   and you **must** not use any other header file except stdio.h.

   For your reference, here are few more examples of encrypted text that you received
   bit earlier but could not decrypt yet.

   ```
   esrevinu olleh
   ```

   ```
   L ssecnirp si eman ym
   ```

3. Write a *non-recursive* function that takes two strings containing only digits $(0, 1, \ldots, 9)$ (30)
   as input and returns sum of those strings as an integer. You **must** not use any other
   header file except stdio.h. You may use the following template to implement it.

   ```
   long int sum_of_two_strings(char str1[], char str2[]) {
           // Complete this function
   }
   ```

   Executing the program looks like the following:

   ```
   Enter string1: 1234
   Enter string2: 11
   The sum of two strings is 1245
   ```

   Hint: ASCII value 48 corresponds to the digit 0.