



תרגיל 1 להגשה במבני נתונים (קורס מס' 10117)

מרצים: ד"ר ראובן חוטובלי וד"ר דוד שטר

תאריך הגשה: 28.11.2020 עד השעה 23:00 . העבודה - בזוגות.

שאלה מס' 1

אפשר לפתור את השאלה בדרכים שונות ואחת מהן:

```
void func (char list1[ ],char list2[ ],char list3[ ], int m, int n)
```

```
{
    if(m <= 0 && n <= 0)
        return;
    else if(m <= 0 && n > 0)
    {
        list3[0] = list2[0];
        func (list1,list2 + 1,list3 + 1,m,n - 1);
    }
    else if(n <= 0 && m > 0)
    {
        list3[0] = list1[0];
        func (list1 + 1,list2,list3 + 1,m - 1,n);
    }
    else
    {
        list3[0] = list1[0];
        list3[1] = list2[0];
        func (list1 + 1,list2 + 1,list3 + 2,m - 1,n - 1);
    }
}
```



שאלה מס' 2

סעיף א'

נתונה קבוצה של n סימנים המשוכנים במערך A שגודלו n .
נגדיר: $f(n,k)$ כמספר האפשרויות לבחירת k סימנים שונים מקבוצת המספרים הנתונה כך שלא יבחרו שני סימנים המשוכנים במערך A במיקומים עוקבים. מהי הנוסחה הרקורסיבית בעבור $f(n,k)$?

$$f(n,k) = f(n-2,k-1) + f(n-1,k) \quad 1.$$

$$f(n,k) = \sum_{k=0}^{n/2} f(n-1,k) \quad 2.$$

$$f(n,k) = 1 + f(n-2,k-1) + 2 + f(n-1,k) \quad 3.$$

$$f(n,k) = 1 + f(n-2,k-1) + f(n-1,k) \quad 4.$$

סעיף ב' נסמן $f(k, n)$ את מספר האפשרויות לפיזור n כדורים זהים לתוך k תאים שונים כך

שכל תא יכיל לכל היותר כדור אחד. נוסחה רקורסיבית בעבור $f(n,k)$ היא:

1.

$$f(k,n) = f(k-1,n) + f(k,n-1)$$

$$f(k,0) = 1 \quad f(1,1) = 1 \quad \text{and} \quad \forall n > 1 \quad f(1,n) = 0 \quad 2.$$

$$f(k,n) = f(k-1,n) + f(k-1,n-1)$$

$$f(k,0) = 1 \quad f(1,1) = 1 \quad \text{and} \quad \forall n > 1 \quad f(1,n) = 0 \quad 3.$$

$$f(k,n) = 1 + f(k-1,n) + f(k-1,n-1)$$

$$f(k,0) = 1 \quad f(1,1) = 1 \quad \text{and} \quad \forall n > 1 \quad f(1,n) = 0$$

4. אף אחת מבין התשובות הנתונות אינה נכונה.

סעיף ג'

מזכירה רשלנית שמה n מכתבים ב- n מעטפות ממוענות, באופן מקרי בלי לקרוא את הכתובות הרשומות על המעטפות. נסמן ב- $f(n)$ את מספר האפשרויות שאף מכתב לא יגיע לתועדתו (לאדם שמיועד לו המכתב). נוסחה רקורסיבית (נסיגה) בעבור $f(n)$ הינה:

$$f(n) = f(n-2) + f(n-1) \quad 1.$$

$$f(n) = n * (f(n-2) + f(n-1)) \quad 2.$$

$$f(n) = (n-1) * (f(n-2) + f(n-1)) \quad 3.$$

4. אף אחת מבין התשובות הנתונות אינה נכונה.



סעיף ד'

לציפי יש n שקלים, ובכל יום היא יכולה לקנות מסטיק ב-1 שקל או שוקולד ב-2 שקלים, או סוכריה ב-2 שקלים.

נגדיר ב- $f(n)$ את מספר הדרכים שבהן יכולה ציפי לבזבז את כל כספה.

מהי הנוסחה הרקורסיבית בעבור $f(n)$?

$$f(n) = f(n-1) + 1 + 2f(n-2) + 4 \quad 1.$$

$$f(n) = \sum_{i=1}^{\frac{n}{2}} f(n-i) \quad 2.$$

$$f(n) = 1 + f(n-1) + f(n-2) \quad 3.$$

$$f(n) = f(n-1) + 2f(n-2) \quad 4.$$

שאלה מס' 3

השלמות בצבע אדום :

```
int what(int n1, int n2)
{
    if (n1<10 && n2<10)
        return (n1==n2) ? 1 : 0;
    if (n1<10)
        return what(n1,n2/10);
    if (n2<10)
        return what(n1/10,n2);
    return what(n1/10,n2/10);
}
```

שאלה מס' 4

```
int makenumintobinary(int num)
{
    if (num == 0) return 0;
    return (makenumintobinary(num / 2))*10 + num % 2;
}
```

שאלה מס' 5

1. $dig1 + '0'$
2. $dig1 - 10 + 'a'$
3. $makenumintohexadecimal(num / 16)$
4. $strlen(s)$
5. $s[len] = dig2$
6. $s[len + 1] = '\0'$



```
#define N 30
void tsearch(int *a, int i, int j, int k);
void main() {
    int a[N] = { 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30
};
    int n, i, k;
    printf("Enter no to search:");
    scanf("%d", &k);
    tsearch(a, 0, N - 1, k);
    getchar();
    getchar();
}

void tsearch(int *a, int i, int j, int k) {
    int m1, m2;
    m1 = i + (j - i) * 1 / 3;
    m2 = i + (j - i) * 2 / 3;
    if (i > j)
    {
        printf("-1");
        return;
    }
    if (k == a[m1])
    {
        printf("\found at %d", m1);
        return;
    }
    else if (k == a[m2])
    {
        printf("\nfound at %d", m2);
        return;
    }
    if (k < a[m1])
        return(tsearch(a, i, m1 - 1, k));
    if (k > a[m2])
        return(tsearch(a, m2 + 1, j, k));
    else
        return(tsearch(a, m1 + 1, m2 - 1, k));
}
```



שאלה מס' 7:

במשחק מסוים משחקים שני שחקנים A ו-B. בתחילת המשחק לשחקן A יש m שקלים, ולשחקן B - n שקלים.

בכל שלב של משחק השחקן שמפסיד משלם שקל אחד למנצח. המשחק יסתיים כאשר אחד השחקנים יפסיד את כל כספו.

כך, למשל, אם בתחילת המשחק ל-A יש 3 שקלים, ול-B 5 שקלים, הסדרה הבאה BBABB מתארת את התוצאות של המשחק (בשלב הראשון והשני המנצח הוא B, בשלב השלישי המנצח הוא A, ובשלב הרביעי והחמישי המנצח הוא B).

	בתום שלב 5	בתום שלב 4	בתום שלב 3	בתום שלב 2	בתום שלב 1	תחילת המשחק
הכסף שלרשות A	0	1	2	1	2	3
הכסף שלרשות B	8	7	6	7	6	5

ברור שבתום חמישה שלבים של המשחק, השחקן A הפסיד והשחקן B מנצח.

זהו סוג של משחק, שבו כל מה שמפסיד שחקן אחד עובר לשחקן השני.

תהי p הסתברות לכך, שהשחקן A ינצח בשלב כלשהו של המשחק.

כתבו נוסחה רקורסיבית (כולל תנאי עצירה) עבור ההסתברות, שהשחקן A יפסיד את כל כספו. רמז לפתרון

נסמן ב- P_m את ההסתברות, שהשחקן A יפסיד במשחק כאשר הוא מתחיל את המשחק עם סכום כסף m שקלים-ואילו השחקן היריב B מתחיל עם n שקלים.

פתרון

ההסתברות שהשחקן A, המתחיל את המשחק ב- m שקלים, יפסיד את המשחק, ב- P_m .

$$P_m = p * P_{m+1} + q * P_{m-1} \quad \text{לכן:}$$

$$P_0 = 1$$

$$P_{m+n} = 0$$

שאלה מס' 8

1. `index == n`
2. `word[index] = 0`
3. `bin(word, n, index + 1)`
4. `word[index] = 1`



שאלה מס' 9:

```
void TR8 (int word[], int n, int index)
{
    if(index == n)
    {
        print_array(word, n);
        return;
    }
    int i;
    for(i = 0; i < K; i++)
    {
        word[index] = i;
        TR8 (word, n, index + 1);
    }
    return;
}
```

שאלה מס' 10

כתבו תכנית רקורסיבית שתקלוט מספר שלם חיובי N (ניתן להניח שיתקבל כזה), ותדפיס את מספר כל האפשרויות של סדרת מספרים כך שסכום איברי הסדרה שווה ל- N ולאחר מכן את כל האפשרויות האלו.
כמו כן התכנית צריכה להדפיס את כל האפשרויות האלו בצורה ממוינת (ראה דוגמה של פלט) דוגמה: עבור קלט 4 הפלט הוא: 8 מספר האפשרויות הוא: 8 והאפשרויות הן:

```
[1, 1, 1, 1]
[1, 1, 2]
[1, 2, 1]
[1, 3]
[2, 1, 1]
[2, 2]
[3, 1]
[4]
```

```
void ex10(int N, int a[], int index)
{
    if (N == 0)
    {
        for (int j = 0; j < index; ++j)
        {
            printf("%d", a[j]);
        }
        printf("\n");
        return;
    }
}
```



```
for (int num = 1; num <= N; ++num)
{
    a[index] = num;
    ex10(N - num, a, index + 1);
}
}
void t_ex10(int N)
{
    int a[10];
    ex10(N, a, 0);
}
int main()
{
    t_ex10(4);
    getchar();
    return 0;
}
```

גרסה אחרת

```
int main(){
    int n = 4; // example of 4
    Q11(n);
}

void Q11(int n){
    int arr[n];
    int num = pow(2,n-1);
    printf("The amount of viable options is %d \n", num);
    // Proved inductively that that's the formula - adding '1' from the left and right,
    // excluding the second printing of '11111...', and adding the last line: 'n'.
    func(n, arr, 0, n);
}

void func(int k, int arr[], int index, int size){
    if(k==0){
        for(int j = 0 ; j < index ; j++)
            printf("%d ", arr[j]);
        printf("\n");
        return;
    }
    if(index == size)
        return;
    for(int i = 1 ; i <= k ; i+=1){
        arr[index] = i;
        func(k - i, arr, index + 1, size);
    }
}
```