



שאלה 5 (20 נקודות)

נתונים שני מערכים של `char` (שימו לב שאלו אינם מחרוזות):

- מערך בשם `arr[]` שאורכו `n`.
- מערך בשם `pat[]` (קיצור של pattern) שאורכו `m`.

השלימו את הפונקציה `printPatterns` (בעמוד הבא) שמקבלת את שני המערכים ואת אורכם, ומדפיסה למסך את כל המופעים של המערך `pat` בתוך המערך `arr`. מופע של המערך `pat` בתוך המערך `arr` איננו חייב להיות רציף.

לדוגמא, עבור הנתונים הבאים: `n = 10, m = 3`

n	b	e	n	b	a	r	e	a	b
---	---	---	---	---	---	---	---	---	---

המערך `arr`:

n	b	a
---	---	---

המערך `pat`:

המופעים של `pat` בתוך המערך `arr` הינם:

n	b	-	-	-	a	-	-	-	-
n	b	-	-	-	-	-	-	a	-
n	-	-	-	b	a	-	-	-	-
n	-	-	-	b	-	-	-	a	-
-	-	-	n	b	a	-	-	-	-
-	-	-	n	b	-	-	-	a	-

```
nb---a----
nb-----a-
n---ba----
n---b---a-
---nba----
---nb---a-
```

עבור הקלט הנ"ל, פלט הפונקציה שלכם צריך להראות כך:

על הפתרון להיות רקורסיבי, ולעבוד בשיטת ה-`backtracking`.

לנוחותכם, ניתן להשתמש בפונקציה העזר הבאה, שמקבלת מערך של `char` בגודל `n` ומדפיסה את תוכנו (אין צורך לממש):

```
void printarray(char arr[], int n);
```

הערות נוספות: שימו לב שהפרמטר `k` המועבר לפונקציה `printPatternsAux` הוא לשימושכם לפי הצורך, ויש יותר מאפשרות אחת נכונה להשתמש בו. כמו כן, שימו לב כי מספר השורות הריקות בקוד לא בהכרח מעיד על מספר שורות הקוד שיש לכתוב, ואורך הקו אינו בהכרח מעיד על אורך השורה שיש להשלים.



```
void printPatterns(char arr[], int n, char pat[], int m)
{
    char *aux; int i;

    aux = (char*)malloc(n);
    for (i=0; i<n; ++i) {
        aux[i] = '-';
    }

    printPatternsAux(arr, n, pat, m, aux, 0);
    free(aux);
}

void printPatternsAux(char arr[], int n, char pat[], int m,
                      char aux[], int k)
{
    if (m==0) {
        printarray(aux, n+k);
        return;
    }

    if (n==0) {
        return;
    }

    if (arr[0] == pat[0]) {
        aux[k] = arr[0];
        printPatternsAux(arr+1, n-1, pat+1, m-1, aux, k+1);
        aux[k] = '-';
    }

    printPatternsAux(arr+1, n-1, pat, m, aux, k+1);
}

}
```