

```
int count_numbers_in_range_O_1(unsigned int help_arr[ ],
                                unsigned int k,
                                unsigned int a,
                                unsigned int b)
{

}

}
```

שאלה 3 (16 נקודות)

נתונים כקלט 2 מערכים, האחד $a[]$ באורך m והשני $b[]$ באורך n כך ש- $m < n$.
א. (6 נקודות) הצע שיטה יעילה לבדוק האם המערך הקצר $a[]$ מופיע כתת מערך של המערך הארוך $b[]$.

דוגמא 1: עבור הקלט:

$a[] = \{3, 5, 2\}$
 $b[] = \{1, 7, 3, 5, 2, 2, 1, 0\}$

Match Found

התוכנית תדפיס:

דוגמא 2: עבור הקלט:

$a[] = \{5, 1, 2\}$
 $b[] = \{1, 7, 3, 5, 2, 2, 1, 0\}$

Did not find a match

התוכנית תדפיס:

```
void find_pattern (int a[ ], unsigned int m, int b[ ], unsigned int n)
{
```

```
}
```

עמוד 6

ב. (2 נקודות) מהי סיבוכיות התכנית שרשמת? נמקו בקצרה.

ג. (8 נקודות) כעת ידוע בנוסף כי:

- המערך b ממין מהקטן לגדול וכל ערך מופיע בו לכל היותר פעם אחת.
- המערך a ממין מהקטן לגדול וכל ערך מופיע בו לכל היותר פעם אחת.

האם ניתן לשפר את סיבוכיות הפונקציה `find_pattern()`?
אם כן – כתבו גירסה יעילה יותר של הפונקציה בעמוד הבא, ורשמו את סיבוכיותה.
אם לא – הסבירו מדוע.

הערה: ניתן להעזר בפונקציית העזר הבאה, `search()`, המחפשת את האיבר c במערך a ממין a , כאשר אזור החיפוש מתחיל באינדקס $start$ ומסתיים ב- end (כולל). הפונקציה מחזירה את האינדקס של מיקום האיבר c במערך, או -1 אם האיבר לא קיים במערך.

```
int search (int a[], int start, int end, int c)
{
    int med;

    if (start==end) {
        if (a[start]==c) return start;
        else return -1;
    }

    med = (start+end)/2;
    if (a[med]<c)
        return search (a, med+1, end, c);
    else
        return search (a, start, med, c);
}
```

```
void find_pattern (int a[ ], unsigned int m, int b[ ], unsigned int n)
{

}

}
```

מה סיבוכיות גרסה זו של הפונקציה? נמקו בקצרה.