

ועדת המשמעת מזהירה!
נבחן שימצאו ברשותו חומרי
עזר אסורים או יתפס בהעתקה
יענש בחומרה עד כדי הרחקתו
מהאוניברסיטה

שאלון סגור

מבוא למדעי מחשב

מבחן מועד ב תשע"ה (2015)

קורס 88-170 / 89-110

מס' מח': 153

שנת: תשע"ה סמסטר: 1 מועד: 2 מטלה: 1
קורס: 89 110 02 מבוא למדעי המחשב

שלוש שעות.

משך הבחינה:

ללא חומרי עזר.

חומר עזר:

הנחיות כלליות:

- המבחן מנוסח בלשון זכר אך מיועד לגברים ו
- חובה לכתוב את כל התשובות בטופס הבחינה.

מחברות הבחינה לא יבדקו !!!

- גודלו של המקום הפנוי לכתיבת התשובה אינו מעיד על אורכה.
- במידה וחסר מקום אפשר לצייר חץ ולכתוב את המשך התשובה מאחורי הדף.
- בשאלות כתיבת קוד ניתן לכתוב הערות בעברית ברורה!
- חובה לענות על כל שאלות המבחן (אין בחירה בין שאלות).
- יש להשתמש אך ורק ב printf ו scanf ל i/o וכן ב free, malloc, calloc, realloc.
- להקצאות ושחרור זיכרון. הקפידו על השימוש הנכון, ואל תשכחו לשחרר זיכרון.
- כדאי לקרוא שאלות עד הסוף לפני שמתחילים לענות.
- ניקוד:

ניקוד	תרגיל
30	1
10	2
15	3
10	4
20	5
15	6

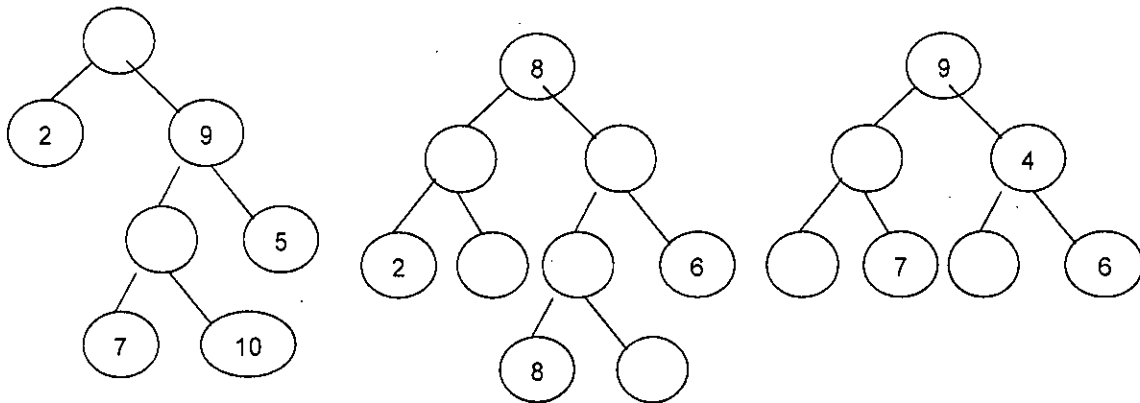
87

בהצלחה!

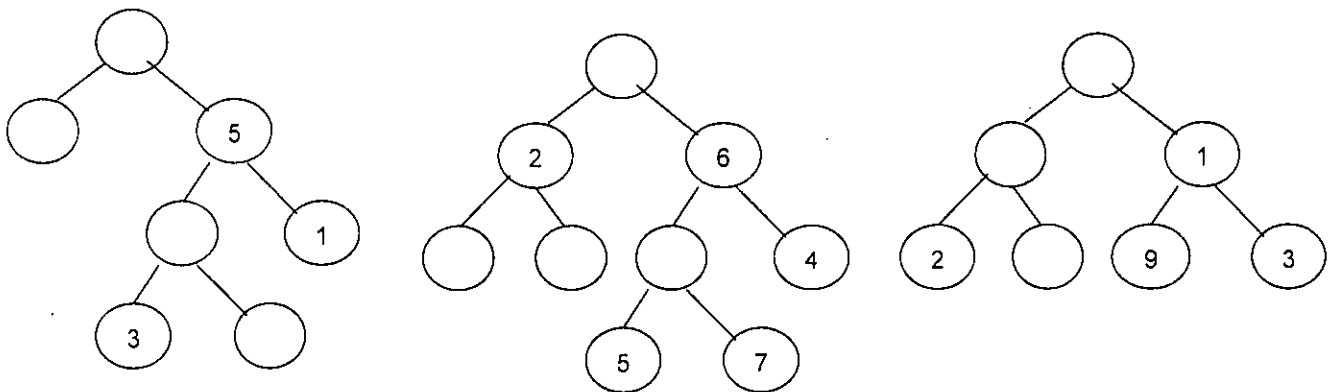


תרגיל 1 (30 נקודות)

עץ חיפוש בינארי הוא עץ בינארי, שבכל קודקוד שלו נמצא מספר שלם, כך ש:
 1. לכל קודקוד, כל הבנים בתת העץ הימני שלו גדולים או שווים לו
 2. לכל קודקוד, כל הבנים בתת העץ השמאלי שלו קטנים ממנו.
 דוגמאות לעצי חיפוש בינאריים:



דוגמאות לעצים שאינם עצי חיפוש בינאריים (חשבו מדוע)



בשאלה זו עליכם להשתמש במימוש הפשוט שראיתם בתרגול לעץ חיפוש בינארי השומר מספר שלם:

```
struct BST_S {
    int val;
    struct BST_S* left;
    struct BST_S* right;
};
```

בכל אחד מן הסעיפים הבאים הקפידו על כללי התכנות הנכון (חלוקה לפונקציות, יעילות וכו').



א. כתבו פונקציה המקבלת מצביע לעץ בינארי ובודקת אם הוא עץ חיפוש בינארי (מחזירה 1 אם כן ו-0 אם לא).

```

int CheckIfBST(struct BST-S* root) {
    // if left and right are null
    if (root->left == NULL && root->right == NULL) // OK
        return 1;

    root->left != NULL
    if ((root->left->val >= root->val) || (root->right->val <= root->val))
        return 0;
    else
        left = CheckIfBST(root->left);
    if ((root->right->val < root->val))
        return 0;
    else
        right = CheckIfBST(root->right);

    if (left == 1 && right == 1)
        return 1;
    else
        return 0;
}

```

Handwritten notes on the left:
 ~left / ~
 NULL
 5/4
 ~right / ~
 ~left / ~
 ~right / ~

Handwritten notes on the right:
 8/
 10

Handwritten notes at the bottom right:
 *root val
 Input: 5, 4, 3, 2, 1
 Output: 1, 2, 3, 4, 5
 Action: ...

ב. כתבו פונקציה שמקבלת מצביע לעץ חיפוש בינארי ואיבר, ומכניסה את האיבר לעץ החיפוש. הפונקציה צריכה להכניס את האיבר להיות עלה בעץ (רמז - מצאו היכן צריך להכניס את האיבר ואז תכניסו אותו)

```

void AddNum(struct BST-S* root, int num) {
    if (root->left == NULL && root->right == NULL) {
        if (num < root->val) {
            root->left = (struct BST-S*) malloc(sizeof(struct BST-S));
            (root->left)->val = num;
            root->left->left = root->left->right = NULL;
        }
        else {
            root->right = (struct BST-S*) malloc(sizeof(struct BST-S));
            (root->right)->val = num;
            (root->right)->right = (root->right)->left = NULL;
        }
    }
    // end of if

    (root->left)->right = (root->left)->left = NULL;
}

```



```

if (num < root->val) {
    AddNum(root->left, num);
    root->left = (struct BST_s*) malloc(sizeof(struct BST_s));
    root->left->val = num;
    root->left->right = (root->left->right) = NULL;
    }
    if (root->right != NULL)
        AddNum(root->right, num);
    else
        root->right = (struct BST_s*) malloc(sizeof(struct BST_s));
        root->right->val = num;
        root->right->right = root->right->left = NULL;
}
//end of outer else
//end of func

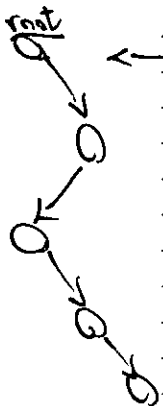
```

ג. עץ חיפוש בינארי נקרא מאוזן, אם יש לו n עלים ועומקו $O(\log n)$. מה סיבוכיות ההכנסה לעץ בינארי מאוזן?

סיבוכיות הכנסה לעץ דו-ענף מאוזן היא $O(\log n)$.
 כיון שמספר הענפים בכל דור, צומח בקצב של $2^{\log n}$, כלומר n עלים.
 כלומר, מספר הענפים בכל דור הוא $2^{\log n}$.
 כלומר, מספר הענפים בכל דור הוא $2^{\log n}$.

ד. מה סיבוכיות ההכנסה לעץ בינארי לא מאוזן? מה המקרה הגרוע ביותר (רמז - תחשבו על הפונקציה שכתבתם)? הדגימו את המקרה הגרוע ביותר עבור עץ בעל חמישה קודקודים.

סיבוכיות הכנסה לעץ דו-ענף לא מאוזן גבוהה (מאד).
 במקרה הגרוע ביותר, נקרא לעץ זה "עץ קווי" (Linked List).
 כלומר, מספר הענפים בכל דור הוא $2^{\log n}$.
 כלומר, מספר הענפים בכל דור הוא $2^{\log n}$.



תרגיל 2 (10 נקודות)

בהינתן שני מספרים a, b שלמים כתבו פונקציה `minus` המחשבת את $a - b$ ללא שימוש באופרטור `-`.
 למשל
`minus(8,2)` מחזיר 6
`minus(9,-3)` מחזיר 12.

על הפונקציה להיות פשוטה ויעילה ככל האפשר (זאת אומרת – מבצעת מספר פעולות מינימלי). יורדו נקודות על מימוש שאינו אופטימלי.

`int a, int b`

```

int minus(8,2) {
    return a + (~b + 1);
}
    
```

✓ 9
10

/* Input: 2 מספרים
 Output: תוצאת בינה
 Action: מחשב את המינוס באמצעות bitshift
 נניח b = 2
 ~b = 1
 ~b + 1 = 2
 a + (-b) = 8 - 2 = 6
 מחזיר 6 */

תרגיל 3 (15 נקודות)

סדרת k-bonachi מוגדרת באופן הבא:
 הערך של כל אחד מ-k האיברים הראשונים הוא 1.
 ערכו של כל איבר הוא סכום של k האיברים הקודמים לו.

למשל

סדרת 2-bonachi היא

... 21 13 8 5 3 2 1 1

סדרת 3-bonachi היא

... 31 17 9 5 3 1 1 1

סדרת 4-bonachi היא

... 25 13 7 4 1 1 1 1

כתבו פונקציה המקבלת n, k , ומחשבת את האיבר ה-n בסדרת k-bonachi.

אתם רשאים לכתוב בפיתון

''' input: k, n - מספרים
 output: האיבר ה-n בסדרת k-bonachi
~~return~~ '''

def kbonacci(k, n):

if $n \leq k$:

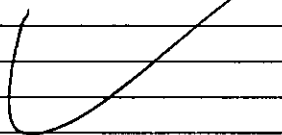
return 1

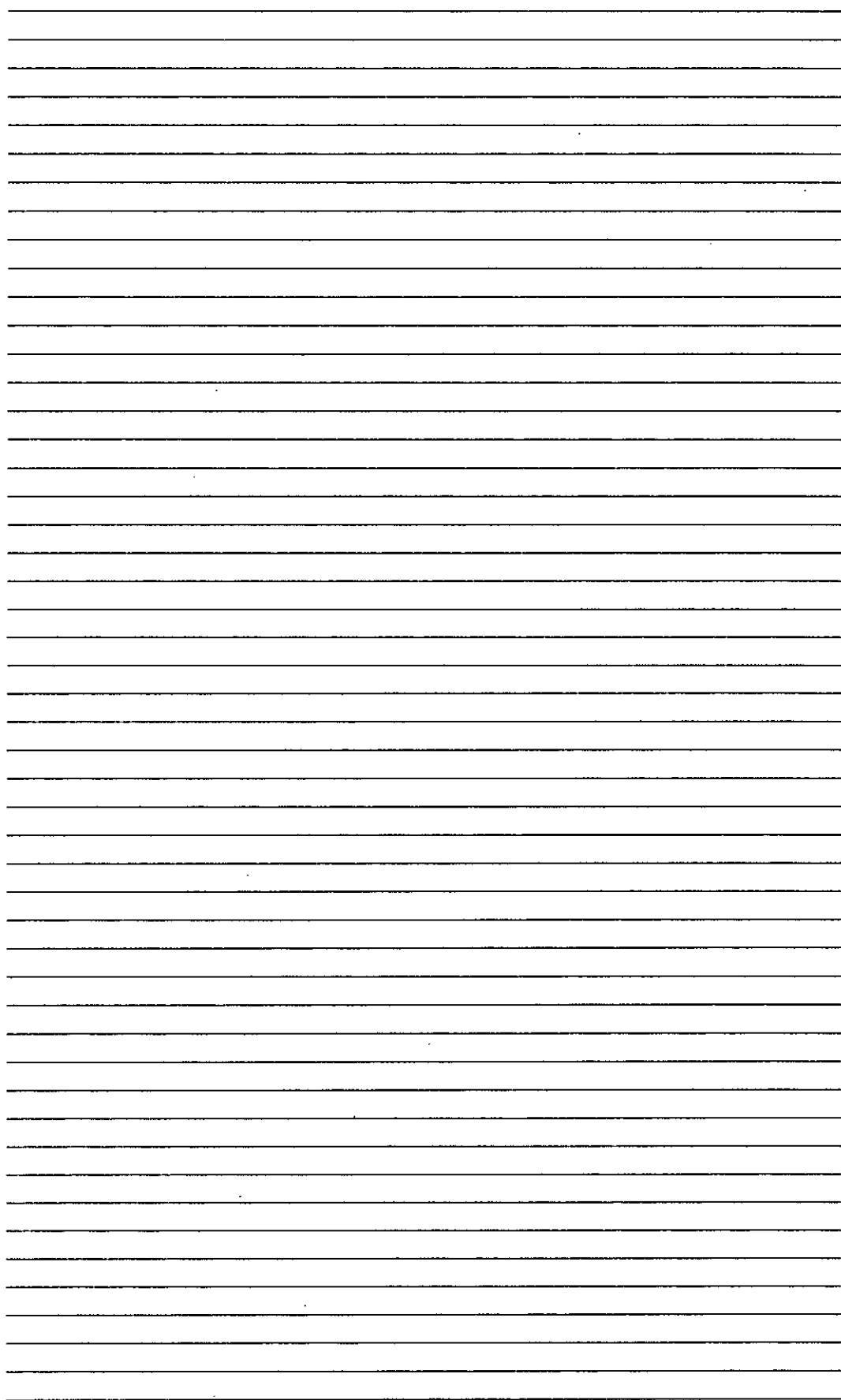
sum = 0

for i in range(1, k+1):

sum += kbonacci(k, n-i)

return sum





תרגיל 4 (10 נקודות)

א. מפתח RSA פומבי הוא זוג N, e . איזה מהערכים הבאים הוא תקין עבור N ? נמק (3 נקודות)

50

לא תקין. $N = p \cdot q$ כאשר p, q ראשוניים. N חייב להיות מכפלה של שני ראשוניים. $50 = 2 \cdot 5 \cdot 5$ אינו מכפלה של שני ראשוניים. 

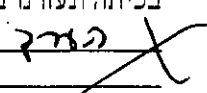
37

לא תקין. 37 ראשוני, לכן N חייב להיות מכפלה של שני ראשוניים שונים. 37 אינו מכפלה של שני ראשוניים שונים. 

77

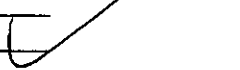
תקין. $N = p \cdot q$, כאשר p, q ראשוניים. $77 = 7 \cdot 11$. 

ב. נתון המפתח הפומבי $N=15, e=3$. מהו הערך של פונקציית אוילר על N (רמז: הפונקציה שחישבנו בכיתה ונעזרנו במשפט פרמה הקטן) (2 נקודות)

הערך של פונקציית אוילר על $N=15$ הוא: $\phi(15) = 8$. 

ג. הצפינו את המספר 7 על ידי המפתח הפומבי המוזכר בסעיף הקודם. (3 נקודות)

~~המספר 7 מוצפן על ידי המפתח הפומבי $N=15, e=3$ ל-343. $7^3 = 343$. $343 \mod 15 = 13$.~~

$E = P^e \pmod{N} = 7^3 \pmod{15} = 343 \pmod{15} = 13$. 

ד. מי מבין הבאים הוא המפתח הסודי? 2, 5, 7, 11, 13. נמקו בקצרה. (2 נקודות)

המפתח הסודי הוא 7. ~~המפתח הסודי הוא 7, 11, 13, 5, 2. המפתח הסודי הוא זה שמתקיים $d \cdot e \equiv 1 \pmod{\phi(N)}$. $\phi(15) = 8$. $7 \cdot 3 = 21 \equiv 1 \pmod{8}$. לכן 7 הוא המפתח הסודי.~~

תרגיל 5 (20 נקודות, 5 נקודות לכל קטע קוד)
עבור כל אחד מקטעי הקוד הבאים, כתוב האם הוא

1. תקין
2. גורם לשגיאת ריצה (או עלול לגרום לשגיאת ריצה)
3. גורם לשגיאת קומפילציה / לינקוג' (נא להסביר)
4. מכיל שגיאה לוגית

בכל מקרה של שגיאה:

1. הקיפו את החלק הבעייתי (בלבד) בקוד.
2. ציינו את סוג השגיאה, והסבירו אותה בקצרה.
3. הסבירו בקצרה כיצד לתקן את השגיאה (ההסבר צריך לכלול מדוע התיקון פותר את הבעיה).

כמו כן, כתבו מה קטע הקוד מדפיס (במקרה של קטע קוד שאינו מתקמפל / רץ קודם תקנו את השגיאה ואז כתבו מה קטע הקוד מדפיס).
כאשר יש

Filename:

Commands

הכוונה היא שהפקודות נמצאות באותו קובץ, כמו שעשינו על הלוח בכיתה (לדוגמא, בחלק מקטעי הקוד יש קבצי h. וכן תוכנה ראשית שעושה להם include)

קטע קוד א'

```
#include<stdio.h>
int main () {
    int i,j = 0;
    for (i = 0; i < 10; i++) {
        printf("%d", i++);
        j = j + i;
        i = j - i;
        j = j - i;
    }
    return 0;
}
```

האם בקטע הקוד קיימת שגיאה? (הקיפו את התשובה):

כן

לא

במידה והתוכנית מדפיסה פלט, כתבו אותו:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9

במידה וסימנחם 'שקיימת שגיאה בקוד':
1. הקיפו בעיגול אך ורק את קטע הקוד הבעייתי.
ציינו את סוג השגיאה, והסבירו אותה בקצרה.

3. הסבירו בקצרה כיצד לתקן את השגיאה:

K

קטע קוד ב'

```
#include<stdio.h>

int main() {
    const int a = 5;
    int i;
    int arr [a];
    for (i = 0; i < 5; i++) {
        arr[i] = i * 2;
    }
    printf("%d", arr[1+2] * arr[0] +
arr[4]);
    return 0;
}
```

האם בקטע הקוד קיימת שגיאה? (הקיפו את התשובה):

כן

לא

במידה והתוכנית מדפיסה פלט, כתבו אותו:

8

במידה וסימנחם שקיימת שגיאה בקוד:
1. הקיפו בעיגול אך ורק את קטע הקוד הבעייתי.
ציינו את סוג השגיאה, והסבירו אותה בקצרה.

3. הסבירו בקצרה כיצד לתקן את השגיאה:



```
#include<stdio.h>
```

```
int i;
```

```
int main(){
```

```
for( ++i; ++i; ++i ) {
```

```
printf("%d", i);
```

```
if(i==4) break;
```

```
}
```

```
return 0;
```

```
}
```

האם בקטע הקוד קיימת שגיאה? (הקיפו את התשובה):

לא

כן

במידה והתוכנית מדפיסה פלט, כתבו אותו:

2,4

במידה וסימנתם שקיימת שגיאה בקוד:

1. הקיפו בעיגול אך ורק את קטע הקוד הבעייתי. ציינו את סוג השגיאה, והסבירו אותה בקצרה.

סוג השגיאה:

~~אם if נראה שהגופה של הקוד
גורמת לבעיה. אך לא ברור כי
רצה להשתמש בלולאה for
יש לכתוב את הקוד הנכון:
for(++i; i; ++i)
במקום לכתוב ++i בלבד.~~

3. הסבירו בקצרה כיצד לתקן את השגיאה:

~~כדי לתקן השגיאה יש לכתוב
את הלולאה כמובן:
for(++i; i; ++i)
במקום לכתוב ++i בלבד
מכאן: 1, 2, 3, 4.~~


```
#include<stdio.h>
```

```
int main() {
    const int i = 1;
    const int *p = &i;
    int j = 2;
    const int *q = &j;
    j = 3;
    printf("%d", *p + *q);
    return 0;
}
```

האם בקטע הקוד קיימת שגיאה? (הקיפו את התשובה):

לא ☒ כן

במידה והתוכנית מדפיסה פלט, כתבו אותו:

במידה וסימנתם שקיימת שגיאה בקוד:

1. הקיפו בעיגול אך ורק את קטע הקוד הבעייתי. ציינו את סוג השגיאה, והסבירו אותה בקצרה.

שגיאה קונפליקט.

הפונקציה q אמור להחזיר לזכרון קבוע, אך משתנה (int) אמורה להחזיר לזכרון המשתנה. אזי מדד קבוע וזמן לפקוד.

3. הסבירו בקצרה כיצד לתקן את השגיאה:

כפי לתקן את השגיאה יש למחוק את הפונקציה המיותרת ש:
int *q = &j;

אחרי התיקון, הוצגו תוצאות: 4.

תרגיל 6 (15 נקודות)
נתונה תוכנית תקינה.

```
#include<stdio.h>

int g();
int f(int n);

int g() {
    int x;
    f(3);
    // הדפיסו את הזיכרון בכל פעם שהתוכנית מגיעה לכאן (***)
    return 2;
}

int f(int n) {
    static int i;
    i+=n;

    if ( n > 3) {
        printf("\ng %d \n", g());
    }

    return i;
}

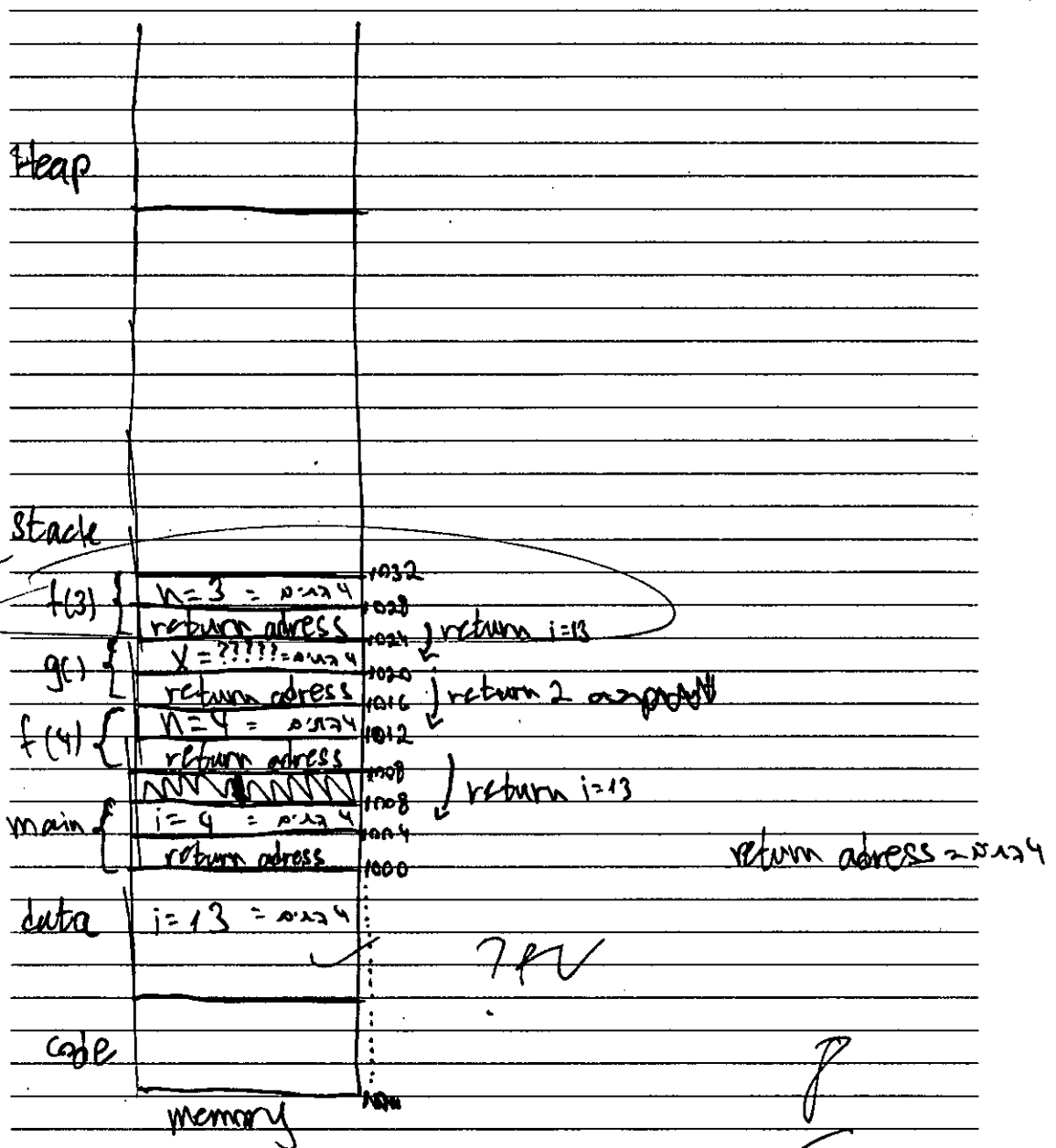
int main () {
    int i;
    for (i = 0; i < 5; i++) {
        printf("\n%d\n",f(i));
    }
    return 0;
}
```

א. הדפיסו את הפלט של התוכנית הבאה (4 נקודות).

0 1 2 3 4
0
1
3
6
9
13
~~16~~

ב. ציירו את תמונת הזיכרון של התוכנית בשלב בו היא מגיעה למקום המסומן ב- (***) . יש להדפיס את הזיכרון, אך ורק במידה והתוכנית מגיעה לשורה. (11 נקודות)







This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There is no text or other markings on the paper.

