

(ב) גמ'ז לטו נכון, הצגה ס לא צב'ז זהה להצגה הקיצ' NULL
שקצ'צ ס'כנ של NULL הזטו ס

(ג) גמ'ז לטו נכון, ג'קן ANSI-ט לטו מח'יק מה יב'י
לדוכ'ז: הפ'ה וא'מ'ז'ים שונים
לדוק'ז של קצ'ז זה "ג'קן ב'י" שג'ה בין ~~מח'יק~~ ~~מח'יק~~
למ'ז-שקצ'צ קצ'ז מח'יקה הפ'ה: האפ'ל זהה.

(ד) גמ'ז א'טו נכון, הקצ'צ'ז בי'נדון למשג'ים לטו נפ'ט
במס'זר ח'צ'ה המ'כ'ית ג'ל'טן ק'מ'ל'כ'ים הקוצ'מ'ים ל'ה.



קריאה ריקורס'ית ל־print_numbers
כל פעם ~~היא~~ ^{עם זריק נמוך} 1-2

רק כאשר גמולת קריאה לפונקציה עם הערך 5 קטן
מ-הקורס' והערות קטנה בסוף ~~הקורס~~ ^{באמצעות} הפקודה 5 .

-3

$\frac{101101101}{101101101}$
 $\frac{101101101}{101101101}$
 $\frac{101101101}{101101101}$

② בקטע 10 שטאט, טאנען, אנהייב, באריי 10

הק'קום א ה סך j דעלס אפס האו j , זא אומר שאלהן מהסאדנס
לא יעדק אים הם ראמים עקום מסוים ודמיה וכן לא יגוס
עסאל הכולל, נמן לנקן שטיב כו על יק' החסר הק'קום $j+1$

שטראטעגיא "ה" מתייחסת קטגוריה 11 המסדק הקו מ"מ" סטקורס
 אומר להכיל ע"ס קטגוריה, ג"כ סדק סקר מייסד ב-ס סוף סדק אטומה
 מייסד ככל סדק קטגוריה ג"כ, סדק סטקורס "ג"כ סדק אטומה קטגוריה
 ג"כ וכן קטגוריה סטקורס קטגוריה סדק אטומה קטגוריה ג"כ
 ה"כ סטקורס = סטקורס קטגוריה ג"כ סדק אטומה קטגוריה ג"כ

ניתן להסידר גמול' if בעל עקומת - הדרג - funcourse
היקום התחילי ורק ~~העבר~~^{העתיד} ~~אולי~~^{או לא} סוכם אולי $\pm cost$.

הערה: סכום כל הספרים בקור כהסטאנטיים טיפוז חזקת גבוה

double k

(א) הקטע 'שלי',
 פשוט ~~הוא~~ ^{הוא} גורם $\&$ הפונקציה malloc גורם
 לקבל מספר קבוע לזיכרון ולכן סוג טיפוס, פקודה
 זו לא מתקבלת, ניתן לבקש שטח בזיכרון וקניי החלפת שורה &
 ?

$$ptrchar = (ptrchar) malloc (sizeof(char*) * 10);$$

~~שטח נוסף גורם & הוא הפניה למשתנה המוגדר const
 לפי המצאת המילה הקטע פסול ptrchar* הוא למעשה const char**
 לא ניתן לשנותו~~

המסדר אינו נכון
 זו אינה בעיה נחשבת

(5)

ניתן לממש כמארג שטח זיכרון
 קבוע את הזיכרון שניתן לבקש
 בהוצאת אובייקט. האלוקציה אינה נחשבת

(10)

```
#define REPLACE_ONEONE(S, Rval, SType) \
{ \
    (SType) temp = (X); \
    Rval = 0; \
    While (temp != 0) \
    { \
        if ((temp & 1) == 1) \
        { \
            if ((temp & 2) == 1) \
            { \
                Rval = Rval << 2; \
                temp = temp >> 2; \
            } \
            Rval = Rval << 1; \
            Rval = Rval | 1; \
            temp >> 1; \
        } \
        Rval = Rval << 1; \
        temp temp = temp >> 1; \
    } \
}
```

S - הוא המספר הנבדק,

Rval - הוא הפרק השמאלית של המספר

SType - הוא סוג המספר S


```

int main ()
{
    int any changed = 0;
    int a1, a2, a3, ra1, ra2, ra3;
    scanf ("please enter three numbers for check, \n %d %d %d, &a1,
           &a2, &a3);

    REPLACE_ONE ONE (a1, ra1, int);
    REPLACE_ONE ONE (a2, ra2, int);
    REPLACE_ONE ONE (a3, ra3, int);
    if (a1 != ra1) /* * */
    {
        printf ("first Number %d was changed to %d", a1, ra1);
        any changed = 1;
    }
    if (a2 != ra2)
    {
        printf ("second number %d was changed to %d", a2, ra2);
        any changed = 1;
    }
    if (a3 != ra3)
    {
        printf ("third second number %d was changed to %d", a3, ra3);
        any changed = 1;
    }
    return 0;
    if (! any change) /* * were zero if it is not changed */
    {
        printf ("None of the numbers where changed");
    }

    return 0;
}

```



```
#define MAX_BIOGRAPHY_NAME 52
#define MAX_WRITE_FOR_BIO 32

struct biography {
    int Id;
    biography* Next;
    char Name[MAX_BIOGRAPHY_NAME];
    char Personality[MAX_WRITE_FOR_BIO];
    int IsHebAvailable;
    int CopyNum;
};
```

```
typedef struct biography biography;
```

```
biography*
void BioListInit ()
{
    int tempid = 0;
    printf("Please enter biography Id to enter the system data, -1 will terminate stop");
    scanf("%d", &tempid);
    while (tempid != -1)
    {
        printf("%s"
```



```

biography* BioList Init ( )
{
    biography* root; /* struct pointer */
    biography* prev, curr bio = malloc (sizeof (biography));
    root = curr bio;
    printf ("please enter biography Id to enter the system data, entering
    (%d) -1 will stop the data adding process");
    scanf ("%d", &(curr bio->Id));
    while ( curr bio->Id != -1) /* -1 is the stop condition */
    {
        /* struct pointer */
        printf ("please enter biography name (%d max characters)",
        MAX_BIOGRAPHY_NAME);
        fgets (&(curr bio->Name[0]), MAX_BIOGRAPHY_NAME, stdin);
        printf ("please enter biography personality (%d max characters)",
        MAX_WRITE_FOR_BIO-3);
        fgets (&(curr bio->Personality[0]), MAX_WRITE_FOR_BIO-2, stdin);
        printf ("please enter 1 if available in hebrew else 0");
        scanf ("%d", &(curr bio->IsHebAvailable));
        printf ("please enter number of copies available");
        scanf ("%d", &(curr bio->Copies Num));
        prev = curr bio;
        curr bio = curr bio->Next;
        curr bio = malloc (sizeof (biography));
        printf ("please enter next biography Id (-1 to stop)");
        scanf ("%d", &(curr bio->Id));
    }
    if ( root->Id == -1) /* if the first node is -1 */
        root = NULL;
    else
        prev->next = NULL; /* the last node is NULL */
    return root;
}

```



```
void Reverse (biography* root)
```

```
{
```

```
    biography* curr, next;
```

/ curr, next הם pointers לביאוגרפיה */*

```
    if ((root != NULL) && (root->next != NULL))
```

```
    {
```

```
        curr = root->next;
```

/ curr = root->next; */*

```
        next = curr->next;
```

```
        root->next = NULL;
```

```
        while (next != NULL)
```

/ while (next != NULL) */*

```
        {
```

```
            curr->next = root;
```

```
            root = curr;
```

```
            curr = next;
```

```
            next = next->next;
```

```
        }
```

```
        curr->next = root;
```

/ curr->next = root; */*

```
        root = curr;
```

```
    }
```

```
    return root;
```

```
int main ( )
```

```
{    biography* curr, next;
```

```
    biography* root = BioListInit();
```

```
    Reverse (root);
```

```
    curr = root;
```

```
    while (curr != NULL)
```

/ while (curr != NULL) */*

```
    {
```

```
        next = curr->next;
```

```
        free (curr);
```

```
        curr = next;
```

```
    }
```

```
    return 0;
```